

(19)



Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number:

0 292 790**A2**

(12)

EUROPEAN PATENT APPLICATION

(21) Application number: 88107596.4

(51) Int. Cl.4: H04L 9/00

(22) Date of filing: 11.05.88

(30) Priority: 29.05.87 US 55502

(43) Date of publication of application:
30.11.88 Bulletin 88/48(84) Designated Contracting States:
DE FR GB IT NL(71) Applicant: International Business Machines
Corporation
Old Orchard Road
Armonk, N.Y. 10504(US)(72) Inventor: Matyas, Stephen Michael, Jr.
8978 Miles Place
Manassa Virginia 22110(US)
Inventor: Meyer, Carl Heinz Wilhelm
27 Norma Court
Kingston New York 12401(US)
Inventor: Bracht, Bruno Oswald
Weinbergstrasse 20
D-7033 Herrenberg(DE)(74) Representative: Grant, Iain Murray
IBM United Kingdom Limited Intellectual
Property Department Hursley Park
Winchester Hampshire SO21 2JN(GB)

(54) Controlling the use of cryptographic keys via generating station established control values.

(57) A method of controlling the use of securely transmitted information in a network of stations in which each potentially cooperating station includes a cryptographic facility which securely stores a master key and in which, for each transmission between a pair of stations, a cryptographic key result is provided for each station of the pair by a generating station which is either one of the pair or a station external to the pair under a cryptographic protocol common to the network, the cryptographic key results for the transmission having a random component notionally particular to the transmission, a master key variant component characteristic of the protocol and a target station component either particular to the stations individually or as a pair, wherein, in response to a generating command invoked in the generating station for establishing a controlled use secure transmission between a designated pair of stations, the generating station generates the cryptographic key result for each designated station, accesses the control value common to the system

for the permitted operation for each of the stations for the particular transmission, combines the control value with the common key result or each individual key result and causes the appropriate combined key result to be established in each station of the pair for the transmission, and wherein the cryptographic facility in each station is arranged, when an operating command is invoked to perform a designated operation with respect to such securely transmitted information, to automatically abort such operation unless it matches the control value.

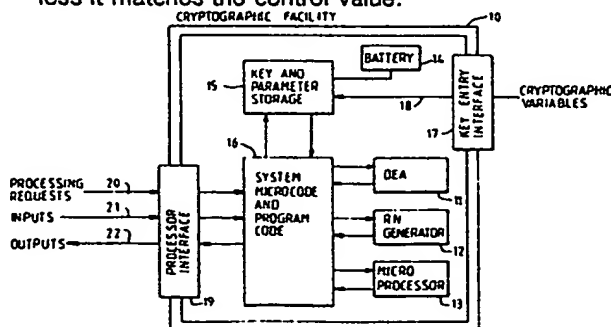


FIG. 2

Best Available Copy

EP 0 292 790 A2

CONTROLLING THE USE OF CRYPTOGRAPHIC KEYS VIA GENERATING STATION ESTABLISHED CONTROL VALUES

The present invention relates to controlling the use of cryptographic keys via generating station established control values. It is noted that generating station may also be a using station.

Cryptography is the only known practical means for protecting information transmitted through a large communications network, be it telephone line, microwave, or satellite. A detailed discussion of how cryptography can be used to achieve communications security is provided in the book by Carl H. Meyer and Stephen M. Matyas entitled Cryptography: A New Dimension in Computer Data Security, John Wiley & Sons (1982). Cryptography can also be used to achieve file security, and a protocol is developed in the Meyer and Matyas book for the encryption of data stored in removable media. Other subjects discussed in the book are enhanced authentication protocols, including personal verification, message authentication, and digital signatures. These subjects are of particular interest to those concerned with electronic funds transfer and credit card applications within the banking and finance industry, or any other area where the originator, timeliness, contents, and intended receiver of a message must be verified.

In the prior art, several references respectively illustrate protocols for distributing cryptographic keys among cryptographically communicating nodes. Further, they discuss authentication as a process independent of the establishment of session keys. These references include U.S. patent No. 4,227,253 to Ehrtam et al. entitled "Cryptographic Communication Security for Multiple Domain Networks" issued October 7, 1980, and U.S. Patent No. 4,218,738 to Matyas et al. entitled "Method for Authenticating the Identity of a User of an Information System" issued August 19, 1980. The Matyas et al. patent involves a node sending a pattern to a terminal requiring the terminal to modify the pattern and remit its modification back to the host to permit a comparison match.

Ehrtam et al., U.S. Patent No. 4,227,253, describe a communication security system providing for the establishment of a session key and the concept of cross-domain keys. The Ehrtam et al. patent typifies a mechanism, i.e., the use of cross-domain keys, used for exchanging session key information between nodes on the one hand the protecting the secrecy of the node master keys on the other hand. More specifically, Ehrtam et al. describe a cryptographic facility at a host computer which, among other things, has a master key KMO with first and second variants of the master key,

denoted KM1 and KM2, and cryptographic operations in support of cryptographic applications and key management, denoted ECPH, DCPH, RFMK, and RTMK. Variants of the master key are obtained by inverting designated bits in the master key to produce different keys, which is just equivalent to Exclusive-ORing predetermined mask values with the master key to produce the variant master keys. The neumonics ECPH, DCPH, RFMK, and RTMK represent the cryptographic operations for Encipher Data, Decipher Data, Reencipher From Master Key, and Reencipher To Master Key. A precise definition of these cryptographic operations is unimportant to the present disclosure; however, the method is such that keys encrypted under KMO can be used beneficially with the ECPH and DCPH functions, keys encrypted under KM1 can be used with the RFMK function, and keys encrypted under KM2 can be used with the RTMK function, but not vice versa. If V0, V1 and V2 denote the mask values which when Exclusive-ORed with KM produce KMO, KM1 and KM2, respectively, then there is an implicit control by the mask values of which cryptographic keys may be beneficially used by which of these cryptographic functions. Although Ehrtam et al. uses variants to control the use of cryptographic keys, by coupling the variants to the cryptographic operations, there is a one-to-one equivalence between the cryptographic operations and the prescribed variants of the key parameters allowed with each cryptographic operation. The Ehrtam et al. architecture does not allow different combinations of variants of keys to be used with each cryptographic function. Thus, for example, if ECPH and DCPH are supported and it is desired to implement data keys with properties of Encipher Only, Decipher Only, and Encipher/Decipher using variants V1, V2 and V3, there is no way to assign these variants to the ECPH and DCPH operations to implement the desired data key properties, i.e., there are not enough variants defined for these operations to accomplish the purpose. In effect, to design such a system requires an ECPH1 which operates with V1, an ECPH2 which operates with V3, a DCPH1 which operates with V2, and a DCPH2 which operates with V4. Therefore, the use of variants to control the use of a cryptographic key in a sophisticated architecture would require the function set to be expanded, and this expansion in the function set has disadvantages the most important of which are the increase of system complexity and cost.

U.S. Patent No. 4,386,233 to Smid et al. entitled "Cryptographic Key Notarization Methods and

Apparatus" issued May 31, 1983, describes a technique of notarizing cryptographic keys for a cryptographic function by encrypting the keys with the cryptographic function using a notarizing cryptographic key derived from identifier designations associated with the encryptor and intended decryptor, respectively, and an interchange key which is accessible only to authorized users of the cryptographic function. In other words, Smid et al. control who can use a key but not how the key can be used. Smid et al's notarizing key is derived by concatenating the binary equivalent of the encryptor's identifier designation with the binary equivalent of the decryptor's identifier designation as an ordered pair and logically combining in an Exclusive-OR operation the concatenated result with the interchange key.

U.S. Patent No. 4,503,287 to Morris et al. entitled "Two-Tiered Communication Security Employing Asymmetric Session Keys" issued March 5, 1985, describes a technique for ensuring communications security between a host computer and another remote computer or terminal by means of a two-tiered cryptographic communications security device and procedure. The Morris et al. technique employs two session keys, one which is encrypted under a master key and transmitted from a remote facility to the host where it is stored, and one which is generated at the host, encrypted under the master key and transmitted to the remote facility where it is used as a session decryptor key.

Thus, while the prior art provides various protocols for distributing cryptographic keys among cryptographically communicating nodes and even provides a way of controlling who may use a cryptographic key at a particular node, there has not been a practical and effective solution to the problem of how to control the use of the cryptographic key at a node, particularly in a sophisticated system. Frequently, different types of keys must be distributed to certain system nodes.

The present invention provides a method of controlling the use of securely transmitted information in a network of stations in which each potentially cooperating station includes a cryptographic facility which securely stores a master key and in which, for each transmission between a pair of stations, a cryptographic key result is provided for each station of the pair by a generating station which is either one of the pair or a station external to the pair under a cryptographic protocol common to the network, the cryptographic key results for the transmission having a random component notionally particular to the transmission, a master key variant component characteristic of the protocol and a target station component either particular to the stations individually or as a pair, wherein, in response to a generating command invoked in the

generating station for establishing a controlled use secure transmission between a designated pair of stations, the generating station generates the cryptographic key result for each designated station, accesses the control value common to the system for the permitted operation for each of the stations for the particular transmission, combines the control value with the common key result or each individual key result and causes the appropriate combined key result to be established in each station of the pair for the transmission, and wherein the cryptographic facility in each station is arranged, when an operating command is invoked to perform a designated operation with respect to such securely transmitted information, to automatically abort such operation unless it matches the control value.

As disclosed herein after, cryptographic techniques according to the present invention are practiced in a communications network having a plurality of stations, each of which has a cryptographic facility which performs cryptographic operations in support of the network encryption function. Such a network can be, for example, an electronic funds transfer (EFT) or point of sale (POS) network and, in any case, would include at least one generating station and at least two using stations. The cryptographic facility at each station in the network has a Key Generation Function (KGF) and a Key Usage Function (KUF). Each key generated by a KGF has an associated control value C which prescribes how the key may be used, and the KUF provides a key authorization function to ensure that a requested usage of a key complies with the control value C.

Two methods may be employed to implement the technique whereby a generating station in a communications network controls the use of a cryptographic key. In the first method, each key and control value are authenticated via a special authentication code before use. In the second method, the key and the control value are coupled during key generation such that the key is recovered only if the correct control value is specified.

In addition to controlling the use of a cryptographic key, the generating station control which generating stations may use a generated and distributed cryptographic key. Two methods are employed to additionally control who may use a cryptographic key. In the first method, each using station has a unique secret transport key shared with a generating station, which the generating station uses to distribute generated data keys to the using stations. Keys are generated by the generating station in such a way that they can be recovered or regenerated only by the designated using stations possessing the correct, designated, secret transport keys. In the second method, each using sta-

tion has a unique nonsecret value associated with it and each pair of using stations share a common, secret transport key with each other and also with the generating station. Keys are generated by the generating station such that they are recovered or regenerated only by the designated using stations possessing the correct, designated, secret transport key. However, since the transport key is shared among two using stations, further cryptographic separation is achieved by using the mentioned public values associated with each using station. Thus, the key generation and recovery procedure is such that the keys distributed to each using station can be recovered or regenerated only by the appropriate using stations possessing the correct, designated, public values. In effect, the transport key ensures that keys prepared for using station i and j cannot be recovered or regenerated at some other using station k, whereas the public values ensure that a key prepared for using station i cannot be recovered or regenerated at using station j, or vice versa.

In summary, four specific cases are described. In the first case, key authentication is used and cryptographic separation is achieved via different, secret transport keys. In the second case, key authentication is used and cryptographic separation is achieved via different public values associated with each using station and via a common, secret transport key. In the third case, no authentication key is used and cryptographic separation is achieved via different, secret transport keys. In the fourth case, no key authentication is used and cryptographic separation is achieved via different public values associated with each using station and via a common, secret transport key.

A control value specifying the usage of a key can be implemented with integrity in three ways:

1. Via authentication codes. The key and control value are distributed separately but are coupled via the authentication code.

2. Via a key distribution function that combines the key with the control value and a secret transport key. Separation is achieved by using different secret transport keys for different using stations.

3. Via a key distribution function that combines the key with the control value, a unique public value associated with the receiving station, and a secret transport key. The transport key is the same for each receiving using station. Separation is achieved via the public value, which is different for each different receiving using station.

The present invention is based on the recognition that additional security benefits could be achieved via a key distribution method where each key had an associated control value that governed how the key would be used by a using station. The

way is provided to couple the key and the control value in a cryptographically secure way to provide a convenient, easy and flexible method of implementing the concept so that keys can be generated at a generating station and distributed to two or more using stations where they can be used in cryptographic operations for cryptographic processing purposes.

An authentication code can be calculated using a secret key which is part of the data being authenticated, whereas in part of the prior art, the message authentication key and the data being authenticated are decoupled. With message authentication, the secret key is used repeatedly to authenticate messages sent from one part to another who share the authentication key, whereas the secret key, as used in this invention, is used just once to authenticate the key itself, the control value, and possible other nonsecret data associated with the key.

Ehram et al, cited above, provides cryptographic separation via two different cross domain keys (or transport keys) for the purpose of distributing a secret dynamically generated key to two using stations. As mentioned above, the variants employed by Ehram et al. provide an implicit control of the beneficial uses to which cryptographic keys may be applied; however, the use of variants severely limit the functions which may be supported. The method according to the present invention of using control values to control the use of cryptographic keys avoids the problems associated with variants since each bit in the control value can be associated with a different cryptographic operation. Thus, if there are 32 cryptographic operations, then a control value of 32 bits or less will cover all possible combinations, whereas with variants this would require potentially 2^{32} different cryptographic operations to allow for all combinations. The invention achieves an obvious economy of scale with the control value over the Ehram et al. method based on variants.

Combining a key and a key variant mask with a secret transport key is a concept described, for example, in copending U.S. Application Serial No. 722,091 filed April 11, 1985, by Walter Ernst Bass et al. for "A Method for Establishing User Authentication with Composite Session Keys Among Cryptographically Communicating Nodes". In that application, the variant of a single cross domain key is used to achieve unidirectionality between two receiving stations, so that reply attacks are thwarted. The unidirectionality feature precludes attacks where certain quantities sent from one point to another as part of establishing a session key cannot beneficially be replayed back to the originating point. The variant mask associated with the cross domain key, in this case, is not a control value as

used in this invention; i.e., it does not specify the usage of the key.

The use of a key distribution function as described in the patent to Smid et al., cited above, describes combining a key to be distributed with the IDs of the sending and receiving nodes and with a secret interchange key. This controls who can use a cryptographic key but does not control how the cryptographic key is to be used. The control value which is used in the present invention, while similar to the concatenation of the sending and receiving IDs as used by Smid et al, is for a wholly different purpose.

The present invention will be described further by way of example with reference to embodiments thereof as illustrated in the accompanying drawings, in which:

Figure 1 is a block diagram illustrating a communication system consisting of a multiplicity of communicating stations connected via a PTT (Post, Telephone and Telegraph) interconnect network;

Figure 2 is a block diagram showing a cryptographic facility capable of encryption/decryption via the Data Encryption Algorithm (DEA);

Figure 3 is a block diagram showing three stations in the network configuration of Figure 1 including a generating station and two using stations;

Figure 4 is a block diagram illustrating the functional relationship between function f_1 and g_1 ;

Figure 6 is a block diagram showing the functional relationship between the functions f_3 and g_3 ;

Figure 7 is a block diagram of the function f_4 ;

Figure 8 is a block diagram showing the functional relationship between the functions f_5 and g_5 ;

Figure 9 is a block diagram showing the functional relationship between the functions f_6 and g_6 ;

Figure 10 is a block diagram of a first embodiment of a generating station wherein a first and second form of a key K are generated via a first function f_1 and a first and second key authentication code are generated via a second function f_2 ;

Figure 11 is a block diagram showing one example of an embodiment for function f_1 ;

Figure 12 is a block diagram showing one example of an embodiment for function f_2 ;

Figure 13 is a block diagram showing a second embodiment of a generating station wherein a first and second form of a key K are generated via a third function f_3 and a first and second key authentication code are generated via a fourth function g_3 related to function f_3 and a fourth function f_4 ;

Figure 14 is a block diagram of one example of an embodiment of the functions f_3 and g_3 ;

Figure 15 is a block diagram of another example of an embodiment of the functions f_3 and g_3 ;

Figure 16 is a block diagram of one example of an embodiment of the function f_4 ;

Figure 17 is a block diagram of another example of an embodiment of the function f_4 ;

Figure 18 is a block diagram showing a first embodiment of a using station related to the first embodiment of a generating station shown in Figure 10 wherein the received key K is recovered via a function g_1 , which is related to the function f_1 , and wherein the received key authentication code is authenticated via the function f_2 ;

Figure 19 is a block diagram showing a second embodiment of a using station related to the second embodiment of the generating station shown in Figure 13 wherein the received key K is recovered via a function g_3 , which is related to function f_3 , and wherein the received key authentication code is authenticated via the function f_4 ;

Figure 20 is a block diagram of a third embodiment of a generating station wherein a first and second form of a key K are generated via a fifth function f_5 ;

Figure 21 is a block diagram of a fourth embodiment of a generating station wherein a first and second form of a key K are generated via a sixth function f_6 ;

Figure 22 is a block diagram of an example of an embodiment of function f_5 and a related function g_5 ;

Figure 23 is a block diagram of an example of an embodiment of function f_5 and a related function g_5 ;

Figure 24 is a block diagram of a third embodiment of a using station related to the third embodiment of the generating station shown in Figure 20 wherein the received key K is recovered via function g_5 , which is related to function f_5 ;

Figure 25 is a block diagram of a fourth embodiment of a using station related to the fourth embodiment of the generating station shown in Figure 21 wherein the received key K is recovered via function g_6 , which is related to function f_6 ; and

Figure 26 is a graphical representation of one possible control vector which may be used.

Referring now to the drawings, and more particularly to Figure 1, a network is shown in which the stations (computers controllers, terminals, and the like) are connected via a PTT (Post, Telephone and Telegraph) interconnect network. Each such station has an encryption/decryption feature capable of end-to-end encryption with any other station in the network. The network referred to here might be an electronic funds transfer (EFT) or point

of sales (POS) network.

Each such station has a cryptographic facility which performs cryptographic operations in support of the network encryption function, such that any station with an implemented cryptographic facility is capable of end-to-end encryption with any other station in the network. The network message formats and protocols necessary to support such cryptographic communication, including those messages necessary to support cryptographic keys and key management functions are not shown here as such messages and protocols are known in the prior art.

Referring now to Figure 2, there is shown a cryptographic facility 10 containing a chip implementation of the Data Encryption Algorithm (DEA) 11, a hardware random number generator 12, a microprocessor 13, a battery 14, a battery-backed random access memory (RAM) 15 for storage of keys and other cryptographic variables, and a memory 16 for storage of system microcode and program code. Keys and cryptographic variables are loaded into the cryptographic facility via a key entry interface 17 and routed to memory 15 via a secure direct path 18. The cryptographic facility can be accessed logically only through inviolate processor interface 19, which is secure against intrusion, circumvention and deception, and which permits processing requests 20 and data inputs 21 to be presented to the cryptographic facility and transformed output 22 to be received from the cryptographic facility.

The cryptographic facility at each station in the network configuration of Figure 1 has a Key Generation Function (KGF) and a Key Usage Function (KUF). Each key generated by a KGF has an associated control value C which prescribes how the key may be used; e.g., encrypt only, decrypt only, generation of message authentication codes, verification of message authentication codes, etc. The KUF provides a key authorization function to ensure that a requested usage of a key complies with the control value C, and it also serves as an authentication function to ensure that a requested key and control value are valid before allowing the key to be used. Thus, the KUF is the logical component of the cryptographic facility that enforces how keys are used at each using station, and in this sense, the KUFs collectively enforce the overall network key usage as dictated by the generating station.

Figure 3 depicts three stations in the network configuration of Figure 1 comprising a generating station and two separate using stations. Each station has a KGF and a KUF, although only the designated generating station has need to exercise the KGF and only the designated using stations have need to exercise the KUF. Thus, it will be

appreciated that any station in the network configuration of Figure 1 can act as a generating station for any other stations acting as using stations, and that a generating station may also act as one of the intended using stations. Moreover, it will be appreciated that this general arrangement can be extended to cover the case where a generating station generates a key in several forms for distribution to several using stations, so that the invention is not limited to only two using stations. All of these combinations and variations are not specifically shown, but it should be evident from the description provided.

Referring again to Figure 3, there is shown a key generated at the generating station via its KGF in a first form with a first control value C and in a second form with a second control value C, which may be the same or different from the first control value C. The generated first form of the key and the first control value are transmitted to a first using station and the generated second form of the key and the second control value are transmitted to a second using station. Thus, the KUF at the first using station permits the received first form of the key to be used only in the manner prescribed by the received first control value and the KUF at the second using station permits the received second form of the key to be used only in the manner prescribed by the received second control value, which may be the same or different from the first control value received by the first using station.

It will be appreciated that each form of the key (first form, second form, etc.) may consist of one or more parameter values representing the information or data necessary to recover, regenerate or reconstitute a previously generated key, and that this process of recovery or regeneration of the key, although not specifically shown in Figure 3, always requires the use of a secret key available to, and known only to the receiving using station. These additional details are described hereinbelow. It will be appreciated still further that additional cryptographic values, beyond those defined as the form of the key and the control value, such as using station unique public value and key authentication code, can be used in addition though such are not all detailed herein. The purpose and use of each of these cryptographic quantities depends on the particular environment and application being considered. The main variations are treated more fully below.

The subject invention may be practiced using two different methods whereby a generating station can control how a cryptographic key may be used at the using station. The first method, which is illustrated by Figures 10 through 19, requires each key and control value, and possibly other key-related data, to be authenticated via a special au-

thentication code before the received, recovered key may be used. The second method, which is illustrated by Figures 20-25, couples the key and control value during the key generation process such that the key is recovered correctly at a using station only if the correct control value has first been specified. Specification of an incorrect control value, in effect, causes a random, unknown key K to be recovered. Thus, if different, incorrect values of C_i and C_j are specified at using station i and j , where there may even be collusion between i and j , the keys recovered by using stations i and j will be spurious (i.e., equal only by pure chance), and hence, no communication between using stations i and j with such incorrectly recovered keys is possible. By exchanging a short verification message which uses the recovered keys, using stations i and j can therefore verify that the key has been correctly recovered before using the key.

Two different methods can be used whereby a generating station can control the using station or stations that may use a distributed cryptographic key. Both methods ensure that a first using station i cannot use or beneficially misuse a key which has been designated for use at another using station j . In the first method, which is illustrated by Figures 10, 11, 12, 18, 20, 22, and 24, cryptographic separation among the keys designated for use at different using stations is accomplished by using different secret transport keys. Each using station shares a different, secret transport key (KR) with each generating station. Thus, at generating station a , transport key KRa_i is used for distribution of key K to using station i , whereas transport key KRa_j is used for distribution of key K to using station j . Under the key distribution procedure, a distributed key K and the transport key are coupled cryptographically such that key K is recovered correctly within the cryptographic facility at using station only if the proper transport key KRa_i has first been initialized. Likewise, key K is recovered correctly within the cryptographic facility at using station j only if the proper transport key KRa_j has first been initialized. In the second method, which is illustrated by Figures 13, 14, 15, 16, 17, 19, 21, 23, and 25, cryptographic separation among the keys designated for use at different using stations is accomplished by assigning and associating a unique, nonsecret value with each using station which is initialized in the cryptographic facility of each respective using station and by sharing a unique, secret transport key among the respective receiving using stations and the generating station. Thus, at generating station a , transport key $KR_{i,j}$ is used for distribution of key K to using stations i and j . The nonsecret or public value associated with each using station is designated PV , so that values PV_i and PV_j would be used for distribution of key K

to using stations i and j , respectively. Under the key distribution procedure, a distributed key K , the public value PV , and the transport KR are coupled cryptographically such that key K is recovered correctly within the cryptographic facility at using station i only if the proper transport key $KR_{i,j}$ and the proper public value PV_i have first been initialized. Likewise, key K is recovered correctly within the cryptographic facility at using station j only if the proper transport key $KR_{i,j}$ and the proper public value PV_j have first been initialized.

From the description above, those skilled in the art will appreciate that Figures 10, 11, 12, and 18 cover the case where key authentication is used and cryptographic separation is achieved via different transport keys; Figures 20, 22 and 24 cover the case where key authentication is used and cryptographic separation is achieved via different public values associated with each using station and a common, secret transport key; Figures 13, 14, 15, 16, 17, and 19 cover the case where no key authentication is used and cryptographic separation is achieved via different transport keys; and Figures 21, 23 and 25 cover the case where no key authentication is used and cryptographic separation is achieved via different public values associated with each using station in conjunction with a common, secret transport key at each using station.

DEFINITION

Several different cryptographic functions are defined. These are designated as functions f_1 , f_2 , f_3 , f_4 , f_5 , f_6 , g_1 , g_3 , g_5 , and g_6 . These functions are used within the cryptographic facility of the generating and using stations for the purposes of key generation, key recovery, and key authentication. A precise definition of each function is given below:

1. Referring to Figure 4, functions f_1 and g_1 are a pair of nonsecret cryptographic functions with the following properties:

a. f_1 and g_1 each have two inputs and one output.

b. The notation $f_1(x,y) = z$ means that z is the output when f_1 is applied to inputs x and y . Likewise, the notation $g_1(x,z) = y$ means that y is the output when g_1 is applied to inputs x and z .

c. f_1 is such that $f_1(x,y)$ depends on each of the inputs x and y ; g_1 is such that $g_1(x,z)$ depends on each of the inputs x and z .

d. f_1 and g_1 are such that if $f_1(x,y) = z$, then $g_1(x,z) = y$. In effect, when the first input parameter of f_1 and g_1 are set equal, then g_1 becomes the inverse of f_1 . Loosely speaking, the first input parameters of f_1 and g_1 are cryptographic keys.

e. $f_1(x,y)$ is easily calculated from x and y . Likewise, $g_1(x,z)$ is easily calculated from x and z .

f. For any given $f_1(x,y) = z$, where z and y are known and x is unknown, it is computationally infeasible to calculate x from y and z . Likewise, for any given $g_1(x,z) = y$, where z and y are known and x is unknown, it is computationally infeasible to calculate x from y and z . With respect to the use of f_1 and g_1 in the present arrangement, this property protects the secrecy of the fixed secret cryptographic key x even if the secret distributed key y should be compromised.

9. For any given $f_1(x,y) = z$, where z is known and x and y are unknown, it is computationally infeasible to calculate y from z . Likewise, for any given $g_1(x,z) = y$, where z is known and x and y are unknown, it is computationally infeasible to calculate y from z . This protects the secrecy of the dynamically distributed secret key y .

h. For any given $f_1(x,y) = z$, where z is known and x and y are unknown, it is computationally infeasible to find a y' and z' , where z' may be equal to z , which satisfy the relationship $f_1(x,y') = z'$. Likewise, for any given $g_1(x,z) = y$, where z is known and x and y are unknown, it is computationally infeasible to find a y' and a z' , where z' may be equal to z , which satisfy the relationship $g_1(x,z') = y'$. This prevents an opponent from forging a dynamically distributed key y that will be accepted by a using station.

2. Referring to Figure 5, function f_2 is a nonsecret cryptographic function with the following properties:

a. f_2 has two inputs and one output.

b. The notation $f_2(x,y) = z$ means that z is the output when f_2 is applied to inputs x and y .

c. f_2 is such that $f_2(x,y)$ depends on each of the inputs x and y .

d. $f_2(x,y)$ is easily calculated from x and y .

e. For any given $f_2(x,y) = z$, where y and z are known and x is unknown, it is computationally infeasible to calculate x from y and z . This protects the secrecy of the dynamically distributed secret key x .

f. For any given $f_2(x,y) = z$, where y and z are known and x is unknown, it is computationally infeasible to find a $y' \neq y$ and a z' , where z' may be equal to z , such that $f_2(x,y') = z'$. This prevents an opponent from forging a control value C and an authentication code that will be properly authenticated and accepted by a using station.

3. Referring to Figure 6, functions f_3 and g_3 are a pair of nonsecret cryptographic functions with the following properties:

a. f_3 and g_3 each have two inputs and one output.

b. The notation $f_3(x,y) = z$ means that z is the output when f_3 is applied to inputs x and y . Likewise, the notation $g_3(x,z) = k$ means that k is the output when g_3 is applied to inputs x and z . The value of k is the dynamically produced secret key being distributed.

c. f_3 is such that $f_3(x,y)$ depends on input y , but may or may not depend on input x . This distinguishes function f_3 from function f_1 .

d. Unlike functions f_1 and g_1 , where $f_1(x,y) = z$ implies that $g_1(x,z) = y$, functions f_3 and g_3 are such that $f_3(x,y) = z$ does not imply that $g_3(x,z) = y$. This property may or may not hold for f_3 and g_3 . The critical feature here is that f_3 and g_3 are less restrictive than functions f_1 and g_1 , whereas, at the same time, they are such that the secret key k can be dynamically produced, distributed and recovered because of the guaranteed functional relationship $g_3(x,f_3(x,y)) = k$. In effect, f_3 and g_3 permit a key distribution using one way functions rather than by using encryption and decryption, which are, by definition, reversible or two way functions.

e. $f_3(x,y)$ is easily calculated from x and y . Likewise, $g_3(x,z)$ is easily calculated from x and z .

f. For any given $f_3(x,y) = z$, where z and y are known and x is unknown, it is computationally infeasible to calculate x from y and z . Likewise, for any given $g_3(x,z) = k$, where z and k are known and x is unknown, it is computationally infeasible to calculate x from k and z . This protects the secrecy of the fixed secret cryptographic key x even if the secret distribution key y should become compromised.

g. For any given $g_3(x,z) = k$, where z is known and x and k are unknown, it is computationally infeasible to calculate k from z . This protects the secrecy of the dynamically distributed secret key k .

h. If function f_3 is such that k can be derived easily from y , or from y and other non-secret data presumed available, then for any given $f_3(x,y) = z$, where z is known and x and y are unknown, it is computationally infeasible to calculate y from z . Again, this protects the secrecy of the dynamically distributed secret key k .

i. For any given $g_3(x,z) = k$, where z is known and x and k are unknown, it is computationally infeasible to find a z' and k' which satisfy the relationship $g_3(x,z') = k'$. This prevents an opponent from forging a dynamically distributed key k that will be accepted by a using station.

j. If function f_3 is such that k can be derived easily from y , or from y and other non-secret data presumed available, then for any given $f_3(x,y) = z$, where z is known and x and y are unknown, it is computationally infeasible to find a y' and z' , where z' may be equal to z , which

satisfy the relationship $f_3(x, y') = z'$. Again, this prevents an opponent from forging a dynamically distributed key k that will be accepted by a using station.

4. Referring to Figure 7, function f_4 is a nonsecret cryptographic function with the following properties:

a. f_4 has three inputs and one output.
b. The notation $f_4(w, x, y) = z$ means that z is the output when f_4 is applied to inputs w , x and y .

c. f_4 is such that $f_4(w, x, y)$ depends on each of the inputs w , x and y .

d. $f_4(w, x, y)$ is easily computed from w , x and y .

e. For any given $f_4(w, x, y) = z$, where x , y and z are known and w is unknown, it is computationally infeasible to calculate w from x , y and z . This protects the secrecy of the dynamically distributed secret key w .

f. For any given $f_4(w, x, y) = z$, where x , y and z are known and w is unknown, it is computationally infeasible to find an x' , y' and z' , where x' or y' or both x' and y' are different from x and y , respectively, which satisfy the relationship $f_4(w, x', y') = z'$. This prevents an opponent from forging a control value C and an authentication code for given public value PV , which will be properly authenticated and accepted by a using station.

5. Referring to Figure 8, functions f_5 and g_5 are a pair of nonsecret cryptographic functions with the following properties:

a. f_5 and g_5 each have three inputs and one output.

b. The notation $f_5(w, x, y) = z$ means that z is the output when f_5 is applied to inputs w , x and y . Likewise, the notation $g_5(w, x, z) = y$ means that y is the output when g_5 is applied to inputs w , x and z .

c. f_5 is such that $f_5(w, x, y)$ depends on each of the inputs w , x and y . g_5 is such that $g_5(w, x, y)$ depends on each of the inputs w , x and z .

d. f_5 and g_5 are such that if $f_5(w, x, y) = z$, then $g_5(w, x, z) = y$. In effect, when the first and second input parameters of f_5 and g_5 are set equal, then g_5 becomes the inverse of f_5 . For practical purposes, the input parameter w in functions f_5 and g_5 is a fixed secret cryptographic key.

e. $f_5(w, x, y)$ is easily calculated from w , x and y . Likewise, $g_5(w, x, y)$ is easily calculated from w , x and z .

f. For any given $f_5(w, x, y) = z$, where z , x and y are known and w is unknown, it is computationally infeasible to calculate w from z , x and y . Likewise, for any given $g_5(w, x, z) = y$, where z , x and y are known and w is unknown, it is computationally infeasible to calculate w from z , x and y .

This protects the secrecy of the fixed secret cryptographic key w even if the secret distributed key y should become compromised.

g. For any given $f_5(w, x, y) = z$, where z and x are known and w and y are unknown, it is computationally infeasible to calculate y from z and x . Likewise, for any given $g_5(w, x, z) = y$, where z and x are known and w and y are unknown, it is computationally infeasible to calculate y from z and x . This protects the secrecy of the dynamically distributed secret key y .

h. For any given $f_5(w, x, y) = z$, where z and x are known and w and y are unknown, it is computationally infeasible to find an x' , y' and z' , where z' may be equal to z but x' or y' or both x' and y' are different from x and y , respectively, which satisfy the relationship $f_5(w, x', y') = z'$. Likewise, for any given $g_5(w, x, z) = y$, where z and x are known and w and y are unknown, it is computationally infeasible to find an x' , y' and z' , where z' may be equal to z but x' or y' or both x' and y' are different from x and y , respectively, which satisfies the relationship $g_5(w, x', z') = y'$. This prevents an opponent from forging a dynamically distributed key y or a control value x or both which would be accepted by a using station.

6. Referring to Figure 9, functions f_6 and g_6 are a pair of nonsecret cryptographic functions with the following properties:

a. f_6 and g_6 each have four inputs and one output.

b. The notation $f_6(v, w, x, y) = z$ means that z is the output when f_6 is applied to inputs v , w , x , and y . Likewise, the notation $g_6(v, w, x, z) = y$ means that y is the output when g_6 is applied to inputs v , w , x , and z .

c. f_6 is such that $f_6(v, w, x, y)$ depends on each of the inputs v , w , x , and y . g_6 is such that $g_6(v, w, x, z)$ depends on each of the inputs v , w , x , and z .

d. f_6 and g_6 are such that if $f_6(v, w, x, y) = z$, then $g_6(v, w, x, z) = y$. In effect, when the first, second and third input parameters of f_6 and g_6 are set equal, then g_6 becomes the inverse of f_6 . For practical purposes, the input parameter v in functions f_6 and g_6 is a fixed secret cryptographic key.

e. $f_6(v, w, x, y)$ is easily calculated from v , w , x and y . Likewise, $g_6(v, w, x, z)$ is easily calculated from v , w , x , and z .

f. For any given $f_6(v, w, x, y) = z$, where z , w , x , and y are known and v is unknown, it is computationally infeasible to calculate v from z , w , x , and y . Likewise, for any given $g_6(v, w, x, z) = y$, where z , w , x , and y are known and v is unknown, it is computationally infeasible to calculate v from z , w , x , and y . This protects the secrecy of the fixed secret cryptographic key w even if the secret distributed key y should become compromised.

g. For any given $f_6(v,w,x,y) = z$, where z , w and x are known and v and y are unknown, it is computationally infeasible to calculate y from z , w and x . Likewise, for any given $g_6(v,w,x,z) = y$, where z , w and x are known and v and y are unknown, it is computationally infeasible to calculate y from z , w and x . This protects the secrecy of the dynamically distributed secret key y .

h. For any given $f_6(v,w,x,y) = z$, where z , y and x are known and v and y are unknown, it is computationally infeasible to find a w' , x' , y' , and z' , where z' may be equal to z but w' or x' or y' or some combination thereof are different from w , x and y , respectively, which satisfy the relationship $f_6(v,w',x',y') = z'$. Likewise for any given $g_6(v,w,x,z) = y$, where z , w and x are known and v and y are unknown, it is computationally infeasible to find a w' , x' , y' and z' , where z' may be equal to z but w' or x' or y' or some combination thereof are different from w , x and y , respectively, which satisfy the relationship $g_6(v,w',x',z') = y'$. This prevents an opponent from forging a dynamically distributed key y or a control value x or both for a given using station with associated public value PV .

i. For any given $g_6(v,w,x,z) = y$, where w , x and z are known and v and y are unknown, it is computationally infeasible to find a w' , w'' , x' , x'' , z' , and z'' , where $w' \neq w''$ and x' and x'' are two different legitimate values of PV , which satisfy the relationship $g_6(v,w',x',z') = g_6(v,w'',x'',z'')$. Note that the value of function g_6 evaluated with inputs v , w' , x' , and z' or with inputs, v , w'' , x'' , and z'' does not need to be known. This property prevents a special type of insider attack where two system users collude to construct alternate inputs with different control values that will allow the same distributed secret key $y' = y''$ to be initialized at two different using stations whose associated public values are x' and x'' , even though the users themselves do not know the value of $y' = y''$. This specific property is needed since the generating station uses the same value of v in function f_6 when calculating the first and second forms of the key to be distributed to two different using stations.

Example embodiments for functions f_1 through f_6 , g_1 , g_3 , g_5 , and g_6 which satisfy the functional definitions given above are provided in Figures 11, 12, 15, 16, 17, 22, and 23. These are described in more detail hereinafter.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Referring now to Figure 10, a first embodiment of a generating station is shown wherein a first and second form of a key K are generated via a first

function f_1 and a first and second key authentication code are generated via a second function f_2 . In Figure 10, there is shown a data base 100 and a cryptographic facility 110 containing a command decoder 115, a random key generator 120, a generate key function 130, a command port 140, an input port 145, and an output port 150. Each using station i shares a unique secret transport key, KR_i , with the generating station, which is used by the generating station to encrypt and forward data keys to that using station. These transport keys, KR_1 , KR_2 , ..., KR_n , are encrypted under a prescribed variant of the master key of the generating station, KM' , and this list of encrypted transport keys, which is indexed by the IDs of the using stations, is stored in data base 100.

Those skilled in the art will understand that the generating station also has a central processing unit (CPU) which manages and controls the key generation process. The CPU (not shown) determines the IDs of the using stations for which keys are to be generated, it determines the control values associated with the keys for each using station, it accesses encrypted transport keys from the data base 100, and it issues generate key commands to the cryptographic facility 110 together with the appropriate control values and encrypted keys. Since CPUs and the processes performed by them in this context are well known in the art, no further description of the CPU and the operations performed by it are needed for an understanding of the present invention.

The steps involved in generating a data key K for using stations i and j can be traced in Figure 10. The CPU first determines that a data key is to be distributed to using stations i and j , i.e., with identifiers ID_i and ID_j , and that the control values at using stations i and j are C_i and C_j , respectively. The identifiers ID_i and ID_j are used via line 160 to access the encrypted transport keys eKM' (KR_i) and eKM' (KR_j), from data base 100, and these encrypted keys are read out on line 165. A "generate key" command on line 170 is input to command port 140 of the cryptographic facility. The encrypted transport keys, eKM' (KR_i) and eKM' (KR_j), and the control values, C_i and C_j , are presented as data inputs at input port 145. In response to the "generate key" command, the command decoder at 115 produces an active generate key function on line 125, which enables the generate key function 130. Once enabled, the generate key function 130 will accept inputs C_j , eKM' (KR_j), C_i , and eKM' (KR_i), from input port 145 and a random data key K from random key generator 120.

The inputs are processed as follows. The value eKM' (KR_i) is decrypted at 131 under master key variant KM' . KM' is a dynamically generated vari-

ant of the master key, KM, where KM is stored in the key and parameter storage of the cryptographic facility 110, as shown in Figure 2, and is available for use by the generate key function 130. The decrypted output KRi and the data key K are processed via combining function f_1 at 133 to produce output $f_1(KR_i, K)$. The data key K and the input control value Ci are processed via combining function f_2 at 134 to produce output $f_2(K, C_i)$. The value $eKM' (KR_i)$ is decrypted at 132 under master key variant KM' . The decrypted output KRj and the data key K are processed via combining function f_1 at 135 to produce output $f_1(KR_j, K)$. The data key K and the input control value Cj are processed via combining function f_2 at 136 to produce output $f_2(K, C_j)$. The four values $f_1(KR_i, K)$, $f_2(K, C_i)$, $f_1(KR_j, K)$, and $f_2(K, C_j)$ are then presented as outputs at output port 150, and appear on lines 151, 152, 153, and 154, respectively.

Those skilled in the art will understand that the serial data represented by $f_1(KR_i, K)$ on line 151 and the serial data $f_2(K, C_i)$ on line 152 are loaded into respective shift registers and are read out in parallel to an output buffer. The output buffer is loaded in parallel with a header and synchronizing data from another register. The data in the output buffer is then read out serially and sent to using station i over a communication link in a conventional manner. In like manner, the serial data represented by $f_1(KR_j, K)$ on line 153 and the serial data $f_2(K, C_j)$ on line 154 are loaded into respective shift registers and are read out into an output buffer. The output buffer has a header and synchronizing data. The data in the output buffer is then read out and sent to using station j. Obviously, the output shift registers and buffer may be multiplexed to sequentially transmit data first to using station i and then to using station j.

Figure 11 shows an example of an embodiment for the function f_1 . As defined, f_1 has two inputs and one output. The inputs K and KR. f_1 comprises an encryption facility E whereby input K is encrypted with KR to provide an output $eKR(K)$.

Figure 12 shows an example of an embodiment for the function f_2 . As defined, f_2 has two inputs and one output. In this case, the inputs are K and C. f_2 comprises an encryption facility E whereby input C is encrypted with K. f_2 also comprises an exclusive OR logic which combines the output of the encryption facility with C to produce the output $eK(C) \oplus C$.

Referring now to Figure 13, a second embodiment of a generating station is shown wherein a first and second form of a key K are generated via a third function f_3 and a first and second key authentication code are generated via a fourth function g_3 , which is related to function f_3 , and a fourth function f_4 . In Figure 13, there is shown a data

base 200 of encrypted keys, a data base 205 of public values, and a cryptographic facility 210 containing a command decoder 215, a random number generator 220, a generate key function 230, a command port 240, an input port 245, and an output port 250. Each pair of using stations i and j that can communicate share a common secret transport key KRij, which is also shared with the generating station. The generating station uses KRij to generate certain cryptovariables which are then sent to using stations i and j. These received cryptovariables are sufficient to allow using stations i and j to regenerate a common data key K. By referring to the combining functions f_3 and g_3 , it will be more fully appreciated that key distribution is accomplished, loosely speaking, via one-way functions instead of using a method of encrypting K at the generating station and decryption to recover K at the receiving using stations. These transport keys, KR1,2, KR1,3,...,KRn,n-1, are encrypted under a prescribed variant of the master key of the generating station, KM' , and this list of encrypted transport keys, which is indexed by the respective IDs of the using stations, is stored in data base 200.

Also associated with each using station i is a public value, PVi. These public values are used to cryptographically distinguish and separate the cryptovariables designated for and transmitted to each respective using station, and the procedure for key distribution is such that the cryptovariables produced and sent to using station i cannot be beneficially used or misused at another using station j. These public values PV1, PV2, ..., PVn, indexed by the ID of the using station, are stored in data base 205.

The generating station also has a central processing unit (CPU) which manages and controls the key generation process. The CPU determines the IDs of the using stations for which keys are to be generated, it determines the control values associated with the keys for each using station, it accesses encrypted keys and public values from the data base, and it issues generate key commands to the cryptographic facility together with the appropriate control values, public values, and encrypted keys.

The steps involved in generating a data key K for using stations i and j can be traced in Figure 13. The CPU first determines that a data key is to be distributed to using stations i and j, i.e., with identifiers IDi and IDj, that the control values at using stations i and j are Ci and Cj, respectively, and that the public values at using stations i and j are PVi and PVj, respectively. The identifiers IDi and IDj are used via line 260 to access the encrypted transport key $eKM' (KR_{ij})$ from data base 200, and this encrypted key is read out on line 261. The identifiers IDi and IDj are also used via line 262 to

access the public values PV_i and PV_j from data base 205, and these public values are read out on line 263. A "generate key" command on line 270 is input to command port 240 of the cryptographic facility 210. The encrypted transport key $eKM' - (KR_{ij})$, the public values PV_i and PV_j , and the control values C_i and C_j are presented as data inputs at input port 245. In response to the "generate key" command, the command decoder 215 produces an active generate key function on line 225, which enables the generate key function 230. Once enabled, the generate key function 230 will accept the inputs $eKM' (KR_{ij})$, PV_j , C_j , PV_i , and C_i from input port 245 and a random number RN from random number generator 220.

The inputs are processed as follows. The value $eKM' (KR_{ij})$ is decrypted at 231 under master key variant KM' . KM' is a dynamically generated variant of the master key, KM , where KM is stored in the key and parameter storage of the cryptographic facility, as shown in Figure 2, and is available for use by the generate key function 230. The decrypted output KR_{ij} and the random number RN are processed via combining function f_3 at 232 to produce output $f_3(KR_{ij}, RN)$. The decrypted output KR_{ij} and the so-produced output $f_3(KR_{ij}, RN)$ are processed via combining function g_3 at 233 to produce output data key K . The data key K , the input control value C_i , and the input public value PV_i are processed via combining function f_4 at 234 to produce output $f_4(K, C_i, PV_i)$, and the data key K , the input control value C_j , and the input public value PV_j are processed via combining function f_4 at 235 to produce output $f_4(K, C_j, PV_j)$. The three values $f_3(KR_{ij}, RN)$, $f_4(K, C_i, PV_i)$, and $f_4(K, C_j, PV_j)$ are then presented as outputs at output port 250 and appear on output lines 251, 252 and 253, respectively.

The serial data represented by $f_4(K, C_i, PV_i)$ on line 252 and the serial data $f_3(KR_{ij}, RN)$ on line 251 are loaded into respective shift registers and are read out in parallel to an output buffer. The output buffer is loaded with a header and synchronizing data from another register. The data in the output buffer is then read out serially and sent to using station i . In like manner, the serial data represented by $f_4(K, C_j, PV_j)$ on line 253 and the serial data $f_3(KR_{ij}, RN)$ on the line 251 are loaded into respective shift registers and are read out in parallel to the output buffer. The output buffer is loaded with a header and synchronizing data from another register. The data in the output buffer is read out serially and transmitted to using station j .

Figure 14 shows an example of an embodiment of the functions f_3 and g_3 . As defined, each of these functions has two inputs and one output. In the case of f_3 , the inputs are KR and RN ; however, the output is a straight through connection of the input RN . In the case of g_3 , the inputs

are again KR and RN . g_3 comprises an encryption facility E in which RN is encrypted under KR . g_3 also comprises an exclusive OR logic which combines the output of the encryption facility E with the input RN to produce the output $eKR(RN) \oplus RN$, where $eKR(RN) \oplus RN$ is defined as the data key K .

Figure 15 shows another example of an embodiment of the functions f_3 and g_3 . In this case, f_3 comprises an encryption facility E which encrypts RN under KR to produce the output $eKR(RN)$. g_3 comprises a decryption facility D which decrypts $eKR(RN)$ under KR to produce RN as the output, where RN is defined as the data key K .

Figure 16 shows an example of an embodiment of the function f_4 . By definition, f_4 has three inputs and one output. The inputs are K , C and PV . f_4 comprises first and second encryption facilities E and exclusive OR logic. The first encryption facility encrypts C under K to produce $eK(C)$ which is combined in the exclusive OR logic with PV to produce $eK(C) \oplus PV$. The output of the exclusive OR logic is encrypted under the second encryption facility to produce to output function $f_4(K, C, PV)$.

Figure 17 shows another example of an embodiment of the function f_4 . In this example, there are three encryption facilities and three exclusive OR logics. The first encryption facility encrypts K under KI , where KI is a fixed nonsecret key, to produce $eKI(K)$ which is exclusive ORed with K to yield $eKI(K) \oplus K$. This output, which will be referred to as $K1$, is used to encrypt C in the second encryption facility, the output of which is exclusive ORed with C to produce $eK1(C) \oplus C$. This output, which will be referred to as $K2$, is in turn used to encrypt PV in the third encryption facility, the output of which is exclusive ORed with PV to provide the output $eK2(PV) \oplus PV = f_4(K, C, PV)$.

Referring now to Figure 18, there is shown a first embodiment of a using station related to the first embodiment of the generating station shown in Figure 10 wherein the received key K is recovered via a function g_1 , which is related to the function f_1 , and wherein the received key authentication code is authenticated via the function f_2 . Figure 18 shows a data base 300 and a cryptographic facility 310 containing a command decoder 315, and "operation allowed" procedure 320, an "abort operation" procedure 327, a check procedure 330, a command port 340, an input port 345, an output port 350, and microcode 380 to perform a requested operation. Each using station i shares a unique secret transport key, KR_i , with the generation station, which is used by the receiving station to receive encrypted data keys from the generating station. The transport keys shared with each generating station are encrypted under a prescribed variant of the master key of the using station, KM' , and these encrypted transport keys are stored in

data base 300. If there were only one generating station and one key shared with that generation station, then there would be only one encrypted transport key in data base 300. The encrypted transport keys in data base 300 are indexed by an identifier, here referred to as "ID of KR", which uniquely identifies each key in the list. Thus, in Figure 18, "ID of KRi" refers to a particular KRi which using station i has shared with the generating station, and is the same KRi used by the generating station to communicate with using station i.

The using station also has a central processing unit (CPU) which manages and controls the key recovery and key usage process. The CPU (not shown) receives a formatted message from the generating station, which contains the ID of the generating station, the IDi of the intended receiving station, the ID of KRi, a control value Ci, a first value $f_1(KRi, K)$, and a second value $f_2(K, Ci)$. The CPU parses messages received from the generating station, extracts data parameters, accesses encrypted transport keys from its data base, and presents key and data parameters to the cryptographic facility 310 in conjunction with requested cryptographic operations.

The steps involved in using a data key K at using station i can be traced in Figure 18. The CPU first determines that a received data key K is to be used in a specific requested cryptographic operation. Using the received value of "ID of KRi", the encrypted transport key $eKM'(KRi)$ is accessed from the data base 300 via line 360, and the encrypted key is read out on line 365. A requested operation on line 370 is input to command port 340 of the cryptographic facility. The encrypted transport key $eKM'(KRi)$ accessed from data base 300, the control value Ci, value $f_1(KRi, K)$, and value $f_2(K, Ci)$ extracted from the received message, and other inputs necessary to the requested cryptographic operation, are presented as data inputs at input port 345. In response to the requested operation, the command decoder 315 activates the "operation allowed" procedure 320. Once enabled, the "operation allowed" procedure 320 will accept inputs $f_2(K, Ci)$, Ci, $f_1(KRi, K)$, and $eKM'(KRi)$ from input port 345. These inputs are temporarily stored in the cryptographic facility 310. Using the just read value of Ci, the "operation allowed" procedure 320 determines whether the usage of data key K in the requested operation is authorized or granted on the basis of data in the control value Ci. If so, then the "operation allowed" procedure 320 produces an activate check procedure on line 325 that enables the check procedure 330. If the use is not authorized, then the "operation allowed" procedure 320 produces an activate abort on line 326. Once enabled, the abort operation 327 erases the inputs

read from input port 345 and temporarily stored in the cryptographic facility 310 and enables another requested operation via command port 340. Once enabled, the check procedure 330 will accept inputs $f_2(K, Ci)$, Ci, $f_1(KRi, K)$, and $eKM'(KRi)$, which have been temporarily stored in the cryptographic facility 310.

The inputs are processed as follows. The value $eKM'(KRi)$ is decrypted at 331 under master key variant KM' . KM' is a dynamically generated variant of the master key KM, where KM is stored in the key and parameter storage of the cryptographic facility 310, as shown in Figure 2, and is available for use by the check procedure 330. The decrypted output KRi and the input value $f_1(KRi, K)$ are processed via combining function g_1 at 332 to produce output data key K. The so-produced data key K and the input control value Ci are processed via combining function f_2 at 333 to produce output $f_2(K, Ci)$. The so-produced value $f_2(K, Ci)$ and the input value $f_2(K, Ci)$ are compared for equality at 334. If not equal, then an activate abort operation is produced on line 328. If equal, then an activate "microcode to perform requested operation" is produced on line 329. Once enabled, the abort operation 327 erases the inputs read from input port 345 and temporarily stored in the cryptographic facility 310 and enables another requested operation via command port 340. Once enabled, the microcode to perform requested operation 380 will accept input to requested operation on line 381 via input port 345 and the so-produced data key K on line 382, which is the output from combining function g_1 at 332. The requested operation is then performed at 380 using these key and data inputs. The output of the requested operation 380 is then presented at output port 250 and appears on line 383.

Referring now to Figure 19, there is shown a second embodiment of a using station related to the second embodiment of the generating station shown in Figure 13 wherein the data key K is regenerated or recovered via a function g_3 , which is related to function f_3 , and wherein the received key authentication code is authenticated via the function f_4 . Figure 19 shows a data base 400 and a cryptographic facility 410 containing a command decoder 415, an "operation allowed" procedure 420, an "abort operation" procedure 427, a check procedure 430, a command port 440, an input port 445, an output port 450, and microcode to perform requested operation 480. Each pair of using stations i and j share a unique secret transport key, KRi, j , which is also shared with the generating station. The transport key KRi, j is used by using station i to recover or regenerate data keys from information received from a generating station, where the so-recovered or so-regenerated data

keys will be used for communication with using station j , and vice versa. The transport keys shared in common with each other using station, and also with the generating station, are encrypted under a prescribed variant of the master key of the receiving station, KM' , and these encrypted transport keys are stored in data base 400. The encrypted transport keys in data base 400 are indexed by an identifier which uniquely relates the key to using station j . Thus, in Figure 19, the term "ID of KR_{ij} " is the identifier of KR_{ij} , which is the transport key that using station i shared with using station j .

The using station also has a central processing unit (CPU) which manages and controls the key recovery and key usage process. The CPU receives a formatted message from the generating station, which contains the ID of the generating station, the ID of the intended using station, the ID of KR_{ij} , a control value C_i , a first value $f_4(K, C_i, PVi)$, and a second value $f_3(KR_{ij}, RN)$. The CPU parses messages received from the generating station, extracts data parameters, accesses encrypted transport keys from its data base, and presents key and data parameters to the cryptographic facility 410 in conjunction with requested cryptographic operations.

The steps involved in using a data key K at using station i can be traced in Figure 19. The CPU first determines that a received data key K is to be used in a specific requested cryptographic operation. Using the received value of "ID of KR_{ij} ", the encrypted transport key $eKM'(KR_{ij})$ is accessed from the data base 400 via line 460, and the encrypted key is read out on line 465. A requested operation on line 470 is input to command port 440 of the cryptographic facility 410. The encrypted transport key $eKM'(KR_{ij})$ accessed from data base 400, the value $f_4(K, C_i, PVi)$, control value C_i , and value $f_3(KR_{ij}, RN)$ extracted from the received message, and other inputs necessary to the requested cryptographic operation but not received in the same message from the generating station, are presented as data inputs at input port 445. In response to the requested operation, the command decoder at 415 activates the "operation allowed" procedure 420. Once enabled, the "operation allowed" procedure 420 will accept inputs $f_4(K, C_i, PVi)$, C_i , $f_3(KR_{ij}, RN)$ and $eKM'(KR_{ij})$ from input port 445. These inputs are temporarily stored in the cryptographic facility 410. Using the just read value of C_i , the "operation allowed" procedure determines whether the usage of data key K in the requested operation is authorized or granted on the basis of data in the control value C_i . If so, then the "operation allowed" produces an activate check procedure on line 425, which enables the check procedure 430. If not, then the "operation allowed" procedure 420 produces an activate abort opera-

tion on line 426. Once enabled, the abort operation 427 erases the inputs read from input port 445 and temporarily stored in the cryptographic facility 410 and enables another requested operation via command port 440. Once enabled, the check procedure 430 will accept inputs $f_4(K, C_i, PVi)$, C_i , $f_3(KR_{ij}, RN)$, and $eKM'(KR_{ij})$ which have been temporarily stored in the cryptographic facility 410.

The inputs are processed as follows. The value $eKM'(KR_{ij})$ is decrypted at 431 under master key variant KM' . KM' is a dynamically generated variant of the master key KM , where KM is stored in the key and parameter storage of the cryptographic facility 410, as shown in Figure 2, and is available for use by the check procedure 430. The decrypted output KR_{ij} and the input value $f_3(KR_{ij}, RN)$ are processed via combining function g_3 at 432 to produce output data key K . The so-produced data key K , the input control value C_i , and public value PVi are processed via combining function f_4 at 433 to produce output $f_4(K, C_i, PVi)$. The public value PVi associated with using station i is stored in the key and parameter storage of the cryptographic facility 410, as shown in Figure 2, and is available for use by the check procedure 430. The so-produced value $f_4(K, C_i, PVi)$ and the input value $f_4(K, C_i, PVi)$ are compared for equality at 434. If not equal, then an activate abort operation is produced on line 428; but if equal, an activate "microcode to perform requested operation" is produced on line 429. Once enabled, the abort operation 427 erases the inputs read from input port 445 and temporarily stored in the cryptographic facility 410 and enables another requested operation via command port 440. Once enabled, the microcode to perform requested operation 480 will accept input to requested operation on line 481 via input port 445 and the so-produced key K on line 482, which is the output from combining function g_3 at 432. The requested operation is then performed at 480 using these key and data inputs. The output of the requested operation 480 is then presented at output port 450 and appears on line 483.

Referring now to Figure 20, there is shown a third embodiment of a generating station wherein a first and second form of a key K are generated via a fifth function f_5 . In Figure 20, there is shown a data base 500 and a cryptographic facility 510 containing a command decoder 515, a random key generator 520, a generate key function 530, a command port 540, an input port 545, and an output port 550. Each using station i shares a unique secret transport key, KR_i , with the generating station, which is used by the generating station to encrypt and forward data keys to that using station. These transport keys, KR_1, KR_2, \dots, KR_n , are encrypted under a prescribed variant of the master key of the generating station, KM' , and this list of

encrypted transport keys, which is indexed by the IDs of the using stations, is stored in data base 500.

The generating station also has a central processing unit (CPU) which manages and controls the key generation process. The CPU determines the IDs of the using station for which keys are to be generated, it determines the control values associated with the keys for each using station, it accesses encrypted keys from the data base, and it issues generate key commands to the cryptographic facility together with the appropriate control values and encrypted keys.

The steps involved in generating a data key K for using stations i and j can be traced in Figure 20. The CPU first determines that a data key is to be distributed to using stations i and j, i.e., with identifiers IDi and IDj, and that the control values at using stations i and j are Ci and Cj, respectively. The identifiers IDi and IDj are used via line 560 to access the encrypted transport keys, $eKM'(KRi)$ and $eKM'(KRj)$, from the data base 500, and these encrypted keys are read out on line 565. A "generate key" command on line 570 is input to command port 540 of the cryptographic facility 510. The encrypted transport keys, $eKM'(KRi)$ and $eKM'(KRj)$, and the control values Ci and Cj, are presented as data inputs at input port 545. In response to the "generate key" command, the command decoder 515 produces an active generate key function on line 525, which enables the generate key function 530. Once enabled, the generate key function 530 will accept inputs Cj, $eKM'(KRj)$, Ci, and $eKM'(KRi)$ from input port 545 and a random data key K from random key generator 520.

The inputs are processed as follows. The value $eKM'(KRi)$ is decrypted at 531 under master key variant KM' . KM' is a dynamically generated variant of the master key KM, where KM is stored in the key and parameter storage of the cryptographic facility 510, as shown in Figure 2, and is available for use by the generate key function 530. The decrypted output KRi, the control value Ci, and the data key K are processed via combining function f_5 at 533 to produce output $f_5(KRi, Ci, K)$. The value $eKM'(KRj)$ is decrypted at 532 under master key variant KM' . The decrypted output KRj, the control value Cj, and K are processed via combining function f_5 at 534 to produce output $f_5(KRj, Cj, K)$. The two values $f_5(KRi, Ci, K)$ and $f_5(KRj, Cj, K)$ are then presented as outputs at output port 550 and appear on output lines 551 and 552, respectively.

The serial data represented by $f_5(KRi, Ci, K)$ on line 531 is loaded into a shift register and read out into an output buffer. The output buffer is also loaded with a header and synchronizing data. The data in the output buffer is then read out serially

and sent to using station i. In like manner, the serial data represented by $f_5(KRj, Cj, K)$ on line 552 is loaded into a shift register and read out into an output buffer. The output buffer is also loaded with a header and synchronizing data. The data in the output buffer is then read out serially and sent to using station j.

Referring now to Figure 21, there is shown a fourth embodiment of a generating station wherein a first and second form of a key K are generated via a sixth function f_6 . In Figure 21, there is shown a data base 600 of encrypted transport keys, a data base of public values 605, and a cryptographic facility 610 containing a command decoder 615, a random key generator 620, a generate key function 630, a command port 640, an input port 645, and an output port 650. Each pair of using stations i and j that can communicate share a common secret transport key $KRij$, which is also shared with the generating station. The generating station uses $KRij$ to encrypt and forward data keys to using stations i and j. These transport keys, $KR1,2$, $KR1,3$, ..., $KRn,n-1$, are encrypted under a prescribed variant of the master key of the generating station, KM' , and this list of encrypted transport keys, which is indexed by the respective IDs of the using stations, is stored in data base 600.

Also associated with each using station i is a public value, PVi , which is used by the generating station to distinguish data keys sent to using station i via transport key $KRij$ from data keys sent to using station j, also via transport key $KRij$. These public values $PV1$, $PV2$, ..., PVn , indexed by the ID of the using station, are stored in data base 605.

The generating station also has a central processing unit (CPU) which manages and controls the key generation process. The CPU determines the IDs of the using stations for which keys are to be generated, it determines the control values associated with the keys for each using station, it accesses encrypted keys and public values from the data base, and it issues generate key commands to the cryptographic facility together with the appropriate control values, public values, and encrypted keys.

The steps involved in generating a data key K for using stations i and j can be traced in Figure 21. The CPU first determines that a data key is to be distributed to using stations i and j, i.e., with identifiers IDi and IDj, that the control values at using stations i and j are Ci and Cj, respectively, and that the public values at using stations i and j are PVi and PVj , respectively. The identifiers IDi and IDj are used via line 660 to access the encrypted transport key $eKM'(KRij)$ from data base 600, and this encrypted key is read out on line 661. The identifiers IDi and IDj are also used via line 662 to access the public values PVi and PVj from data

base 605, and these public values are read out on line 663. A "generate key" command on line 670 is input to command port 640 of the cryptographic facility 610. The encrypted transport key $eKM' - (KRij)$, the public values PVi and PVj , and the control values Ci and Cj are presented as data inputs at input port 645. In response to the "generate key" command, the command decoder 615 produces an active generate key function on line 625, which enables the generate key function 630. Once enabled, the generate key function 630 will accept inputs $eKM' (KRij)$, PVj , Cj , PVi , and Ci from input port 645 and a random data key K from random key generator 620.

The inputs are processed as follows. The value $eKM' (KRij)$ is decrypted at 631 under master key variant KM' . KM' is a dynamically generated variant of the master key KM , where KM is stored in the key and parameter storage of the cryptographic facility 610, as shown in Figure 2, and is available for use by the generate key function 630. The decrypted output $KRij$, the control value Ci , the public value PVi , and the random data key K are processed via combining function f_6 at 632 to produce output $f_6(KRij, Ci, PVi, K)$. The decrypted output $KRij$, the control value Cj , the public value PVj , and the random data key K are processed via combining function f_6 at 633 to produce output $f_6(KRij, Cj, PVj, K)$. The two values $f_6(KRij, Ci, PVi, K)$ and $f_6(KRij, Cj, PVj, K)$ are then presented as outputs at output port 650 and appear on lines 651 and 652, respectively.

The serial data represented by $f_6(KRij, Ci, PVi, K)$ on line 651 is loaded into a shift register and read out in parallel to an output buffer. The output buffer is also loaded with a header and synchronizing data. The data in the output buffer is then read out serially and sent to using station i . In like manner, the serial data represented by $f_6(KRij, Cj, PVj, K)$ on line 652 is loaded into a shift register and read out in parallel to the output buffer. The output buffer is also loaded with a header and synchronizing data. The data in the output buffer is then read out serially and sent to using station j .

Figure 22 shows an example of an embodiment of function f_5 and a related function g_5 . By definition, these functions have three inputs and one output. In the case of f_5 , the inputs are KR , C and K . f_5 comprises first and second encryption facilities. In the first encryption facility, K is encrypted under C to produce $eC(K)$. This is in turn encrypted under KR in the second encryption facility to produce the $eKR(eC(K)) = f_5(KR, C, K)$. In the case of g_5 , the inputs are C , KR and $f_5(KR, C, K) = Y$. g_5 comprises first and second decryption facilities. In the first decryption facility Y is decrypted under KR to produce $dKR(Y)$. This is in turn decrypted under C in the second decryption facility

to produce $g_5(KR, C, Y) = K$.

Figure 23 shows an example of an embodiment of function f_6 and a related function g_6 . By definition, each has four inputs and one output. The inputs to f_6 are KR , PV , C , and K . f_6 comprises first, second and third encryption facilities. In the first encryption facility, K is encrypted under C to produce $eC(K)$. This is in turn encrypted under PV in the second encryption facility to produce $ePV(eC(K))$. Finally, the output of the second encryption facility is encrypted under KR in the third encryption facility to produce $f_6(KR, C, PV, K) = Y$. The inputs to g_6 are C , PV , KR , and Y . g_6 comprises first, second and third decryption facilities. In the first decryption facility, Y is decrypted under KR to produce $dKR(Y)$. This is in turn decrypted under PV in the second decryption facility to produce $dPV(dKR(Y))$. Finally, the output of the second decryption facility is decrypted under C to produce the output $g_6(KR, C, PV, Y) = K$.

Referring now to Figure 24, there is shown a third embodiment of a using station related to the third embodiment of the generating station shown in Figure 20 wherein the received key K is recovered via function g_5 , which is related to function g_5 . In Figure 24, there is shown a data base 700 and a cryptographic facility 710 containing a command decoder 715, an "operation allowed" procedure 720, an "abort operation" procedure 727, a key recovery function 730, a command port 740, an input port 745, an output port 750, and microcode 780 to perform requested operation. Each using station i shares a unique secret transport key, KRi , with the generating station, which is used by the receiving station to receive encrypted data keys from the generating station. The transport keys shared with each generating station are encrypted under a prescribed variant of the master key of the receiving station, Km' , and these encrypted transport keys are stored in data base 700. If there were only one generating station and one key shared with that generating station, then there would be only one encrypted transport key in data base 700. The encrypted transport keys in data base 700 are indexed by an identifier, here referred to as "ID of KR ", which uniquely identifies each key in the list. Thus, in Figure 24, "ID of KR " refers to a particular KRi which using station i has shared with the generating station, and is the same KRi used by the generating station to communicate with using station i .

The using station also has a central processing unit (CPU) which manages and controls the key recovery and key usage process. The CPU receives a formatted message from the generating station, which contains the ID of the generating station, the ID of the intended using station, the ID of KRi , a control value Ci , and a value $f_5(KRi, Ci, K)$.

The CPU parses messages received from the generating station, extracts data parameters, accesses encrypted transport keys from its data base, and present key and data parameters to the cryptographic facility in conjunction with requested cryptographic operations.

The steps involved in using a data key K at using station i can be traced in Figure 24. The CPU first determines that a received data key K is to be used in a specified requested cryptographic operation. Using the received value of "ID of KRi", the encrypted transport key $eKM'(KRi)$ is accessed from the data base 700 via line 760, and the encrypted key is read out on line 765. A requested operation on line 770 is input to command port 740 of the cryptographic facility 710. The encrypted transport key $eKM'(KRi)$ accessed from data base 700, the control value Ci and the value $f5(KRi, Ci, K)$ from the received message, and other inputs necessary to the requested cryptographic operation but not received in the same message from the generating station, are presented as data inputs at input port 745. In response to the requested operation, the command decoder 715 activates the "operation allowed" procedure 720. Once enabled, the "operation allowed" procedure 720 will accept inputs $f5(KRi, Ci, K)$, Ci and $eKM'(KRi)$ from input port 745. These inputs are temporarily stored in the cryptographic facility 710. Using the just read value of Ci , the "operation allowed" procedure 720 determines whether the usage of data key K in the requested operation is authorized or granted on the basis of data in the control value Ci . If so, then the "operation allowed" procedure 720 produces an activate key recovery function on line 725, which enables the key recovery function 730. If not, then the "operation allowed" procedure 720 produces an activate abort operation on line 726. Once enabled, the abort operation 727 erases the inputs read from input port 745 and temporarily stored in the cryptographic facility 710 and enables another requested operation via command port 740. Once enabled, the key recovery function 730 will accept inputs $f5(KRi, Ci, K)$, Ci and $eKM'(KRi)$, which have been temporarily stored in the cryptographic facility.

The inputs are processed as follows. The value $eKM'(KRi)$ is decrypted at 731 under master key variant KM' . KM' is a dynamically generated variant of the master key, KM , where KM is stored in the key and parameter storage of the cryptographic facility 710, as shown in Figure 2, and is available for use by the key recovery function 730. The decrypted output KRi and the input values of Ci and $f5(KRi, Ci, K)$ are processed via combining function g_5 at 732 to produce output data key K. The successful completion of the combining function g_5 also raises an activate operation on line 729, which

enables the microcode that performs the requested operation. Once enabled, the microcode to perform requested operation 780 will accept input to requested operation on line 781 via input port 745 and the so-processed data key K on line 782, which is input from combining function g_5 at 732. The requested operation is then performed at 780 using these key and data inputs. The output of the requested operation 780 is then presented at output port 750 and appears on line 783.

Referring now to Figure 25, there is shown a fourth embodiment of a using station related to the fourth embodiment of the generating station shown in Figure 21 wherein the received key K is recovered via function g_6 , which is related to function f_6 . In Figure 25, there is shown a data base 800 and a cryptographic facility 810 containing a command decoder 815, an "operation allowed" procedure 820, an "abort operation" procedure 827, a key recovery function 830, a command port 840, an input port 845, an output port 850, and microcode 870 to perform requested operation. Each pair of using stations i and j share a unique secret transport key, $KRij$, which is also shared with the generating station. The transport key $KRij$ is used by using station i to recover or regenerate data keys from information received from a generating station, where the so-recovered or so-regenerated data keys will be used for communication with using station j, and vice versa. The transport keys shared in common with each other using station, and also with the generating station, are encrypted under a prescribed variant of the master key of the receiving station, KM' , and these encrypted transport keys are stored in data base 800. The encrypted transport keys in data base 800 are indexed by an identifier which uniquely relates the key to using station j. Thus, in Figure 25, the term "ID of $KRij$ " is the identifier of $KRij$, which is the transport key that using station i shares with using station j.

The using station also has a central processing unit (CPU) which manages and controls the key recovery and key usage process. The CPU receives a formatted message from the generating station, which contains the ID of the generating station, the IDi of the intended using station, the ID of $KRij$, a control value Ci , and a value $f6(KRij, Ci, PVi, K)$. The CPU parses messages received from the generating station, extracts data parameters, accesses encrypted transport keys from its data base, and presents key and data parameters to the cryptographic facility in conjunction with requested cryptographic operations.

The steps involved in using a data key K at using station i can be traced in Figure 25. The CPU first determines that a received data key K is to be used in a specific requested cryptographic operation. Using the received value of "ID of $KRij$ ", the

encrypted transport key eKM' ($KRij$) is accessed from the data base 800 via line 860, and the encrypted key is read out on line 865. A requested operation on line 870 is input to command port 840 of the cryptographic facility 810. The encrypted transport key eKM' ($KRij$) accessed from data base 800, the value $f6(KRij, Ci, PVi, K)$ and control value Ci extracted from the received message, and other inputs necessary to the requested cryptographic operation but not received in the same message from the generating station, are presented as data inputs at input port 845. In response to the requested operation, the command decoder 815 activates the "operation allowed" procedure 820. Once enabled, the "operation allowed" procedure 820 will accept inputs $f6(KRij, Ci, PVi, K)$, Ci , and eKM' ($KRij$) from input port 845. These inputs are temporarily stored in the cryptographic facility 810. Using the just read value of Ci , the "operation allowed" procedure 820 determines whether the usage of data key K in the requested operation is authorized or granted on the basis of data in the control value Ci . If so, then the "operation allowed" procedure 820 produces an activate key recovery function on line 825, which enables the key recovery function 830. If not, then the "operation allowed" procedure 820 produces an activate abort operation on line 826. Once enabled, the abort operation 827 erases the inputs read from input port 845 and temporarily stored in the cryptographic facility 810 and enables another requested operation via command port 840. Once enabled, the key recovery function 830 will accept inputs $f6(KRij, Ci, PVi, K)$, Ci and eKM' ($KRij$), which have been temporarily stored in cryptographic facility 810.

The inputs are processed as follows. The value eKM' ($KRij$) is decrypted at 831 under master key variant KM' . KM' is a dynamically generated variant of the master key, KM , where KM is stored in the key and parameter storage of the cryptographic facility 810, as shown in Figure 2, and is available for use by the key recovery function 830. The decrypted output $KRij$, the input value Ci , the value PVi , and the input value $f6(KRij, Ci, PVi, K)$ are processed via combining function g_6 at 832 to produce output data key K . The public value PVi associated with using station i is stored in the key and parameter storage of the cryptographic facility 810, as shown in Figure 2, and is available for use by the key recovery function 830. The successful completion of the combining function g_6 also raises an activate operation on line 829, which enables the microcode that performs the requested operation. Once enabled, the microcode 880 to perform the requested operation will accept input to requested operation on line 881 via input port 845 and the so-produced data key K on line 882, which

is the output from combining function g_6 at 832. The requested operation is then performed at 880 using these key and data inputs. The output of the requested operation 880 is then presented at output port 850 and appears on line 883.

At a using station we have shown the recovery and controlled use of a single data key. However, the invention could be enlarged to provide for the simultaneous recovery and controlled use of any number of any type of keys at each using station. The public values $PV1...PVn$ could be the IDs or functions of the IDs of the respective using stations, in which case the two data bases would be combined. The public values could also be public keys in an RSA algorithm type system or could simply be random numbers. The control value may, under certain protocols, reside at the using station so that it would not be necessary to transmit the control value to the designated using stations. For example, a using station may send with its request for a cryptographic key from the generating station an appropriate or specified control value.

Figure 26 illustrates but one possible control vector which may be used. The control vector may be viewed as a bit map of one dimension which represents the control value used in any of the several embodiments of the invention described above. From left to right in Figure 26, the first and second bits may be ones or zeros to control whether the cryptographic key can be used to encipher or decipher or both encipher and decipher data. Following the first two bits is an initial chaining value (ICV) which, under the DES algorithm, controls the mode of block chaining used in the process. The values for the ICV are shown in Figure 26 and they are mutually exclusive. Next, are two bits which control whether the cryptographic key can be used for message authentication code generation (MACGEN) or verification (MACVER). Following that is an ICV for the message authentication code. Next are two bits which control the translation of cipher text to or from another form. This is followed by an ICV for the translation of cipher text. Finally, there are two bits which control the translation of a personal identification number (PIN).

To summarize, the sender, by specifying how a particular key should be handled at the receiver, via the control block C, determines key management operations at the receiver. Consequently, exposures at the receiver are minimized, provided that the integrity of the cryptographic operations are assured, e.g., via a cryptographic facility.

Claims

1. A method of controlling the use of securely transmitted information in a network of stations in which each potentially cooperating station includes a cryptographic facility which securely stores a master key and in which, for each transmission between a pair of stations, a cryptographic key result is provided for each station of the pair by a generating station which is either one of the pair or a station external to the pair under a cryptographic protocol common to the network, the cryptographic key results for the transmission having a random component notionally particular to the transmission, a master key variant component characteristic of the protocol and a target station component either particular to the stations individually or as a pair, wherein, in response to a generating command invoked in the generating station for establishing a control use secure transmission between a designated pair of stations, the generating station generates the cryptographic key result for each designated station, accesses the control value common to the system for the permitted operation for each of the stations for the particular transmission, combines the control value with the common key result or each individual key result and causes the appropriate combined key result to be established in each station of the pair of the transmission, and wherein the cryptographic facility in each station is arranged, when an operating command is invoked to perform a designated operation with respect to such securely transmitted information, to automatically abort such operation unless it matches the control value.

2. A method as claimed in claim 1, wherein either each station has both a key generation function and a key usage function, the combined key result being generated by the key generation function of one of a pair of stations and transmitted to its own key usage function and to the key usage function of the other station; or a server station for the network has a key generation function for the network, the remaining user stations having key usage functions and the combined key result is generated in the server station and is transmitted to the key usage functions of a designated pair of stations.

3. A method as claimed in claim 2, wherein each station stores a data base including a plurality of encrypted secret transport keys unique to each using station and indexed by identifications of the using stations, the encrypted secret transport keys being encrypted under a variant of the master key;

the server station or any such station in response to a generating command, generating a random key in its cryptographic facility as a cryptographic key;

accessing the encrypted secret transport keys for the designated using stations using the identification for the using stations;

decrypting in the cryptographic facility of the generating station the accessed secret transport keys for the designated using stations using the variant of the master key;

combining in the cryptographic facility of the generating station the decrypted secret transport keys with generated cryptographic key to produce a combined function f_1 for each designated using station;

reading the control value for the permitted operation for each designated using station;

combining the generated cryptographic key with the control value for each designated using station to produce a combined function f_2 ; and

transmitting the combined functions f_1 and f_2 for each designated using station and the control value to the corresponding designated using station.

4. A method as claimed in claim 3, wherein the combining operation to produce the combined function f_1 is performed by encrypting the generated cryptographic key under the decrypted secret transport keys for each designated using station, and wherein the combining operation to produce the combined function f_2 is performed by first encrypting the control values for each designated using station under the generated cryptographic key and then exclusive ORing the thus encrypted control values with the control values for each designated using station.

5. A method as claimed in claim 3 or claim 4, comprising at a designated using station in response to the requesting of a cryptographic operation requiring the use of the cryptographic key generated by the generating station in combination with a control value;

accessing the encrypted secret transport key and temporarily storing in the local cryptographic facility, such encrypted secret transport key together with the control value and the combined functions f_1 and f_2 ,

checking, in the local cryptographic facility, the control value to determine if the requested operation is allowed by such control value;

if the requested operation is allowed, decrypting the encrypted secret transport key using a variant of said master key, combining the decrypted secret transport key with the combined function f_1 using a combining function g_1 to recover the generated cryptographic key, combining the recovered cryptographic key with the control value to produce an authentication function f_2 , comparing the temporarily stored combined function f_2 , and if the stored combined function f_2 and the authenticating function f_2 are equal, enabling the request-

ed cryptographic operation; otherwise,

aborting the requested cryptographic operation and erasing the values temporarily stored in the cryptographic facility of the using station.

6. A method as claimed in claim 5 wherein the combining function g_1 is an inverse function of the function f_1 .

7. A method as claimed in claim 2 wherein each station stores, in a first data base, a plurality of encrypted secret transport keys unique to each pair of using stations in the network and indexed by identifications of pairs of using stations sharing a secret transport key, the encrypted secret transport keys being encrypted under a variant of the master key, and in a second data base, a plurality of nonsecret values unique to each using station in the network and indexed by identifications of such using stations;

the server station or each such station, in response to a generating command, generating a random number in its cryptographic facility;

accessing the encrypted secret transport keys shared by designated using stations using the identifications for the using station pairs sharing the encrypted secret transport keys;

accessing the nonsecret values for the designated using stations using the identifications for the designated using stations;

decrypting the accessed secret transport keys using the variant of the master key;

combining the generated random number with the decrypted secret transport keys to produce a combined function f_3 for each of the designated using stations;

combining the decrypted secret transport key with said combined function f_3 to generate the cryptographic key;

reading the control value for the permitted operation for each designated using station;

for each designated using station, combining the generated cryptographic key with the control value and the nonsecret value for the designated using station to produce a combined function f_4 for the designated using station; and

transmitting the combined functions f_3 and f_4 for each designated using station and the control value to the corresponding designated using station.

8. A method as claimed in claim 7, wherein the combining to produce the combined function f_3 is performed by encrypting the random number under the decrypted secret transport key for each designated using station.

9. A method as claimed in claim 7, or claim 8 wherein the combining to produce the combined function f_4 is performed by first encrypting the control values under the generated cryptographic

key and then exclusive ORing the thus encrypted control values with the nonsecret values for each designated using station.

10. A method as claimed in any of claims 7 to 9 further comprising, at a designated using station in response to the requesting of a cryptographic operation requiring the use of the cryptographic key generated by the generating station in combination with a control value;

accessing the encrypted secret transport key and temporarily storing in the local cryptographic facility the encrypted secret transport key together with the control value and the combined functions f_3 and f_4 transmitted from the generating station;

checking the control value to determine if the requested operation is allowed by a such control value;

if the requested operation is allowed, decrypting the stored encrypted secret transport key using a variant of the master key, combining the decrypted secret transport key with the combined function f_3 using a combining function g_3 to recover the generated cryptographic key, combining the recovered cryptographic key with the control value and the nonsecret value for the designated using station to produce an authentication function f_4 , comparing the temporarily stored combined function f_4 with the authenticating function f_4 , and if such stored combined function f_4 and authenticating function f_4 are equal, enabling the requested cryptographic operation; otherwise,

aborting the requested cryptographic operation and erasing the temporarily stored values.

11. A method as claimed in claim 10 wherein the combining to produce the combined function f_3 is performed by making the combined function f_3 equal to the random number and wherein the combining function g_3 involves encrypting the random number under the decrypted secret transport key and then exclusive ORing the encrypted random number with the random number.

12. A method as claimed in the claim 10 wherein the combining function g_3 is an inverse function of the function f_3 .

13. A method as claimed in claim 2, wherein each station stores a data base including a plurality of encrypted secret transport keys unique to each of the using stations and indexed by identifications of the using stations, the encrypted secret transport keys being encrypted under a variant of the master key;

the server station or each such station in response to a generating command, generating a random number in its cryptographic facility as a cryptographic key;

accessing the encrypted secret transport keys for the designated using stations using the identification for such using stations;

decrypting the accessed secret transport keys for the designated using stations using the variant of the master key;

reading the control value for the permitted operation for each designated using station;

combining the decrypted secret transport keys with the generated cryptographic key and the control values for each designated using station to produce combined function f_5 for each designated using station; and

transmitting the combined function f_5 for each designated using station and the control value to the corresponding designated using station.

14. A method as claimed in claim 13 wherein the combining operation to produce the combined function f_5 for each designated using station is performed by encrypting the generated cryptographic key under the control value for the corresponding designated using station to produce a first encrypted value and then encrypting such first encrypted value under the decrypted secret transport key for the corresponding designated using station.

15. A method as claimed in claim 13 or claim 14, further comprising at a designated using station, in response to the requesting of a cryptographic operation requiring the use of the cryptographic key generated by the generating station in combination with a control value;

accessing the encrypted secret transport key and temporarily storing in the cryptographic facility of the designated using station the encrypted secret transport key together with the control value and the combined function f_5 ;

checking the control value to determine if the requested operation is allowed by said control value;

if the requested operation is allowed, decrypting the said encrypted secret transport key using a variant of the master key, combining the decrypted secret transport key with the combined function f_5 and the control value using a combining function g_5 to recover the generated cryptographic key, and enabling the requested cryptographic operation; otherwise,

aborting the requested cryptographic operation and erasing the temporarily stored values.

16. A method as claimed in claim 15 wherein the combining function g_5 is an inverse function of the function f_5 .

17. A method as claimed in claim 2, wherein each station stores, in a first data base, a plurality of encrypted secret transport keys unique to each pair of using stations in the network and indexed by identifications of pairs of using stations sharing a secret transport key, the encrypted secret transport keys being encrypted under a variant of the master key, and

in a second data base, a plurality of nonsecret values unique to each using station in the network and indexed by identifications of the using stations;

the server station or each such station, in response to a generating command, generating a random number in its cryptographic facility as a cryptographic key;

accessing the encrypted secret transport keys shared by designated using stations using the identifications for the using station pairs sharing the encrypted secret transport keys;

accessing the nonsecret values for the designated using stations using the identifications for such designated using stations;

decrypting the accessed secret transport keys using the variant of the master key;

reading the control value for the permitted operation for each designated using station;

combining the generated cryptographic key with the decrypted secret transport key, control value and nonsecret value for each designated using station to produce a combined function f_6 for each designated using station; and

transmitting the combined function f_6 for each designated using station and the control value to the corresponding designated using station.

18. A method as claimed in claim 17 wherein the combining operation to produce the combined function f_6 is performed by encrypting the cryptographic key under the control value for the designated using station to produce a first encrypted value, encrypting the first encrypted value under the nonsecret value for the designated using station to produce a second encrypted value, and encrypting the second encrypted value under the decrypted secret transport key for the designated using station.

19. A method as claimed in claim 17 or claim 18, further comprising at a designated using station in response to the requesting of a cryptographic operation requiring the use of the cryptographic key generated by the generating station in combination with a control value;

accessing the encrypted secret transport key and the nonsecret value and temporarily storing in the cryptographic facility of the designated using station the transmitted encrypted secret transport key together with the control value and the combined function f_6 ;

checking the control value to determine if the requested operation is allowed by such control value;

if the requested operation is allowed, decrypting the stored encrypted secret transport key using a variant of the master key, combining the decrypted secret transport key with the control value, the nonsecret value and the combined function f_6 using a combining function g_6 to recover the

generated cryptographic key, and enabling the requested cryptographic operation; otherwise,

aborting the requested cryptographic operation and erasing the temporarily stored values.

20. A method as claimed in claim 18, wherein the combining function g_6 is an inverse function of the function of f_6 .

5

10

15

20

25

30

35

40

45

50

55

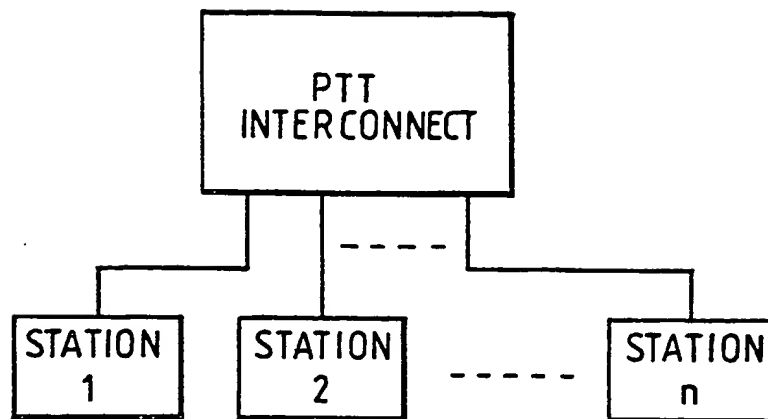


FIG. 1

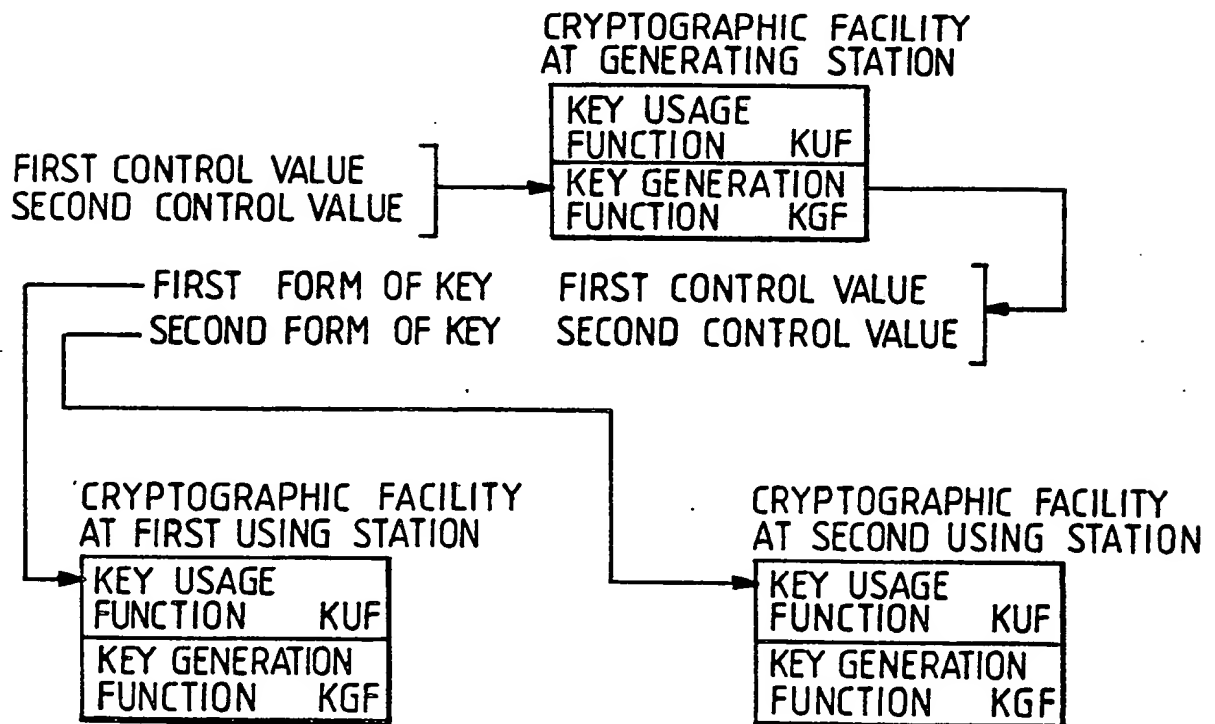


FIG. 3

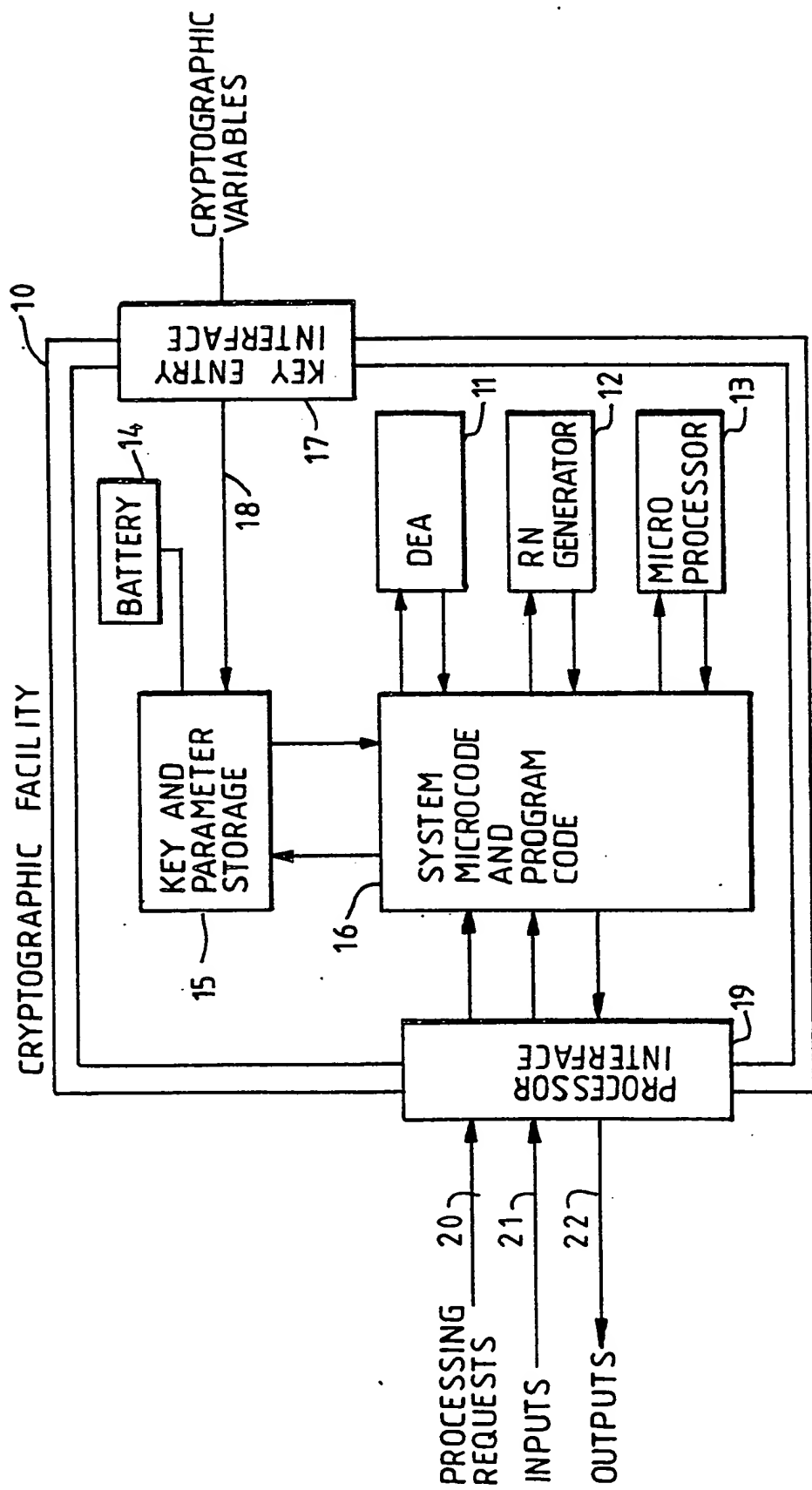
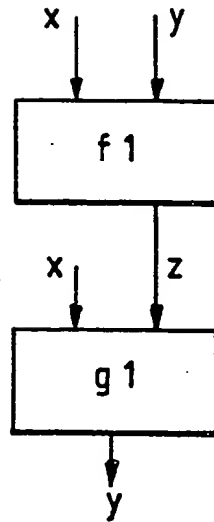
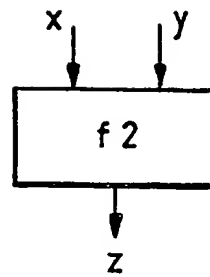
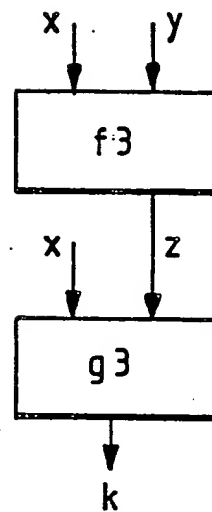


FIG. 2

FIG. 4FIG. 5FIG. 6

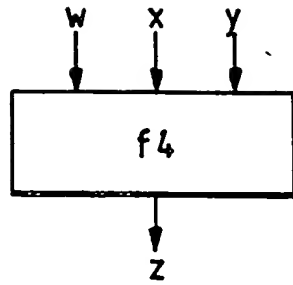


FIG. 7

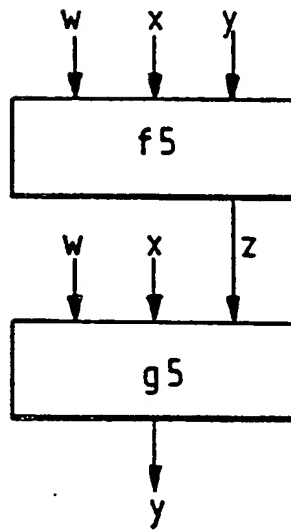


FIG. 8

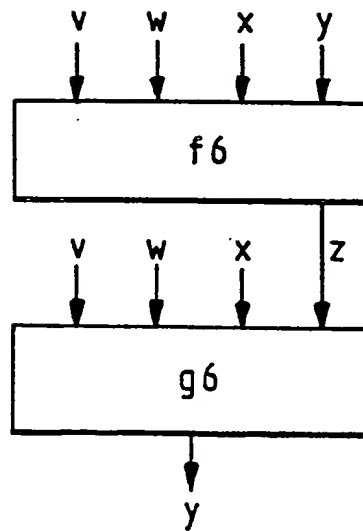
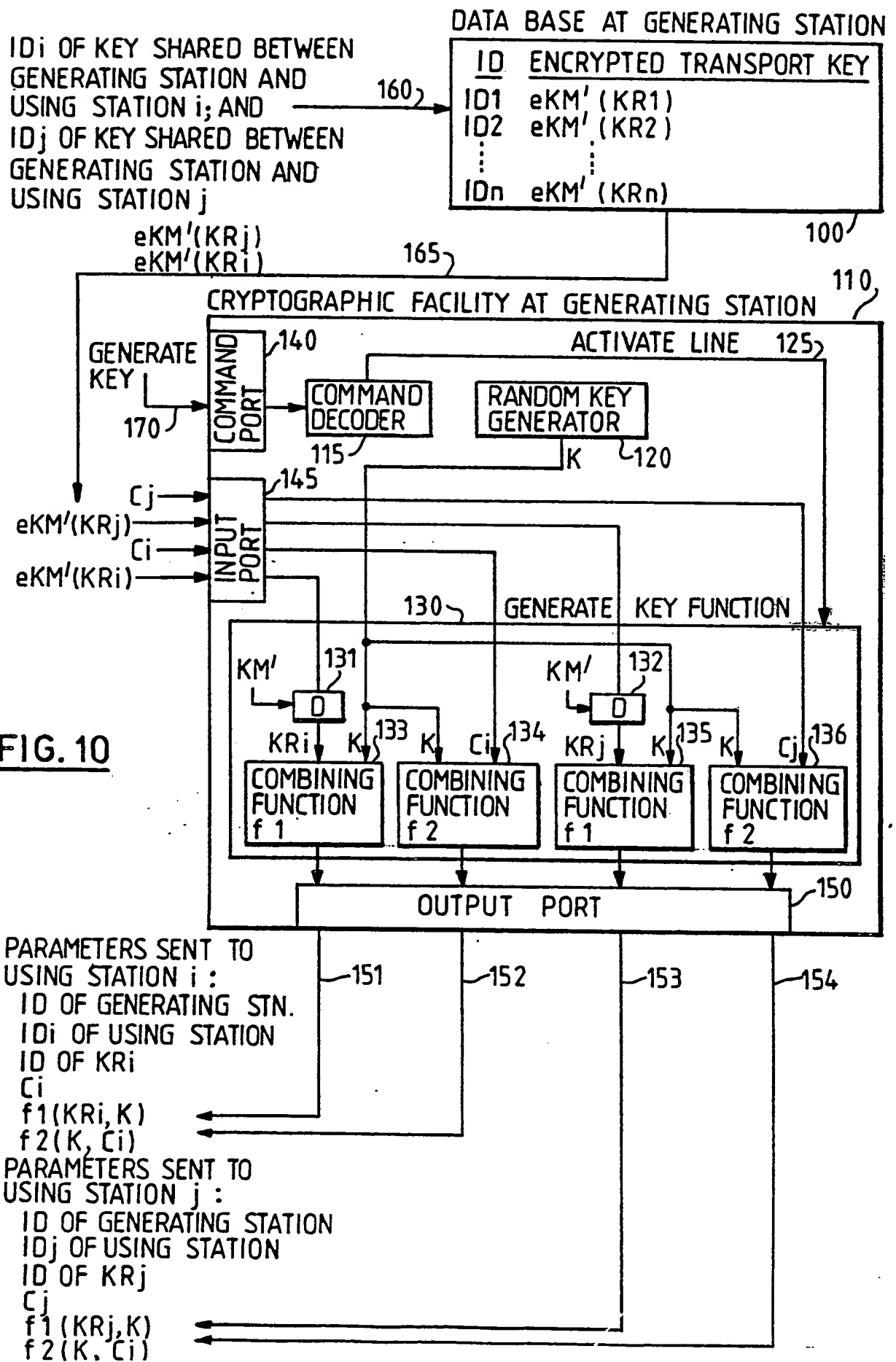
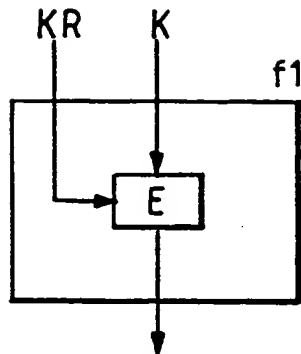


FIG. 9



INPUTS:



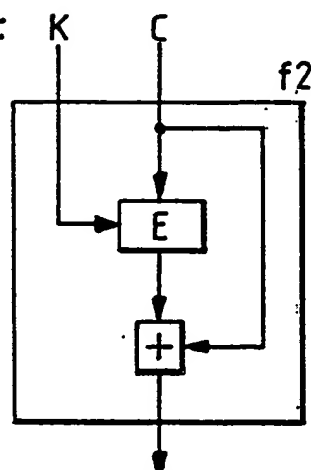
OUTPUT:

$$eK(K) = f1(KR, K)$$

FIG. 11

EXAMPLE OF FUNCTION $f1$

INPUTS:

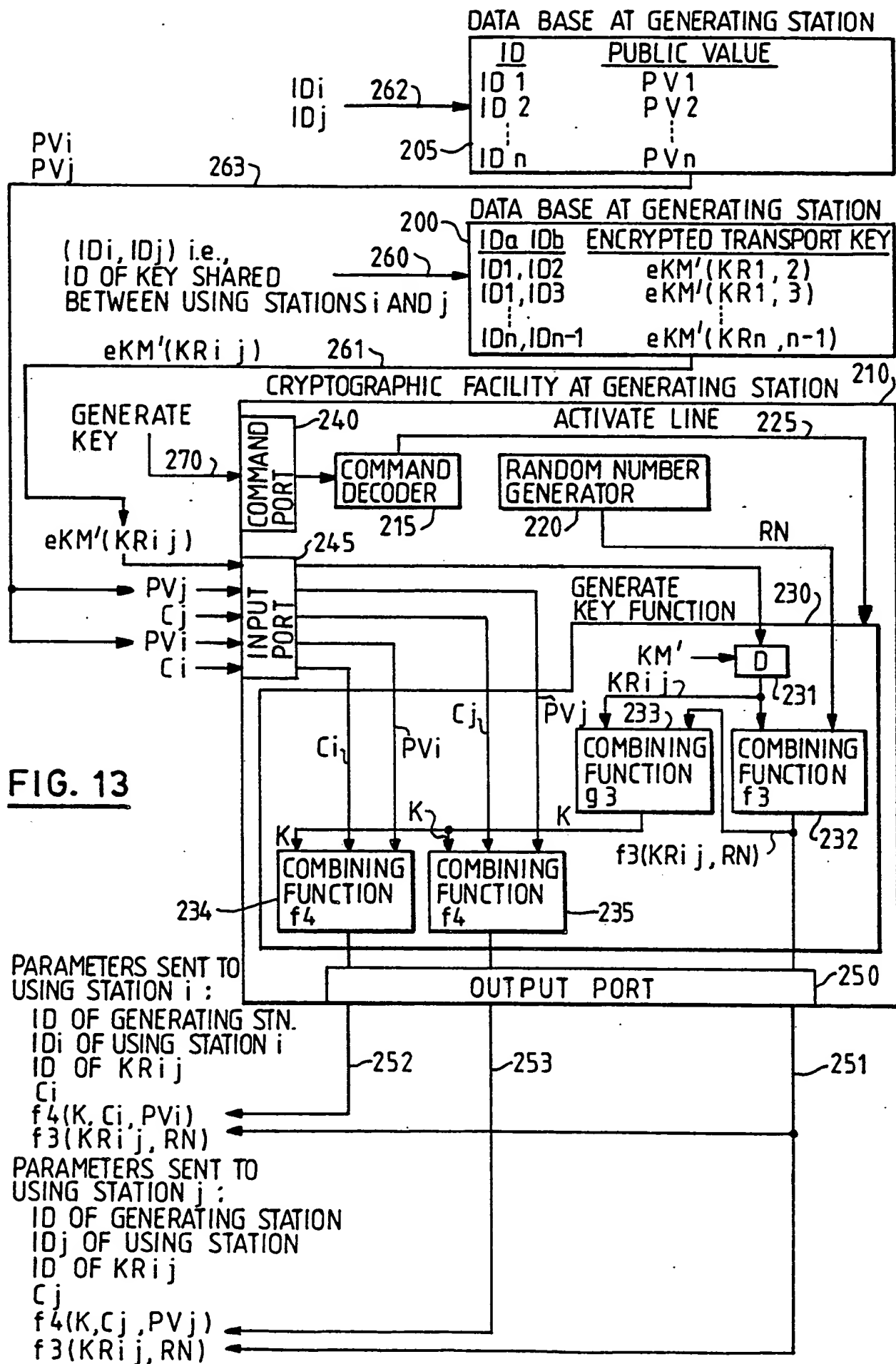


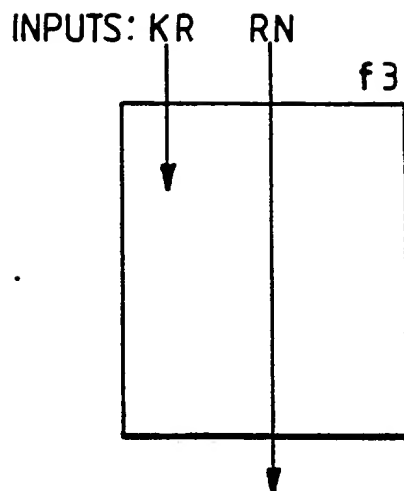
OUTPUT:

$$eK(C) + C = f2(K, C)$$

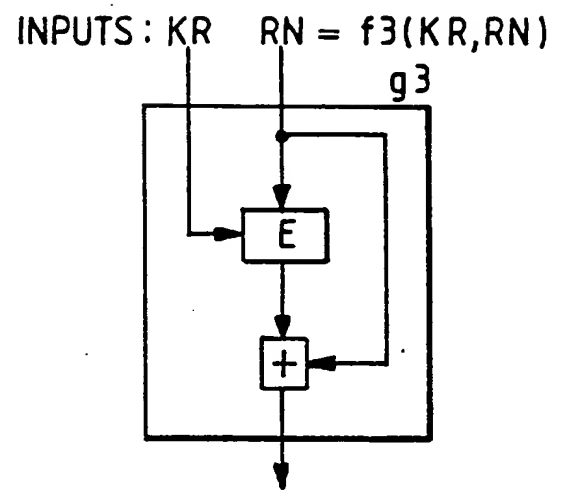
FIG. 12

EXAMPLE OF FUNCTION $f2$



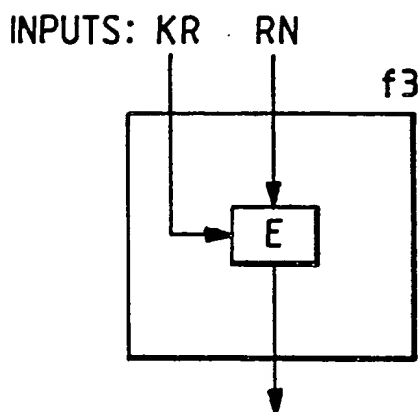


OUTPUT: $RN = f3(KR, RN)$

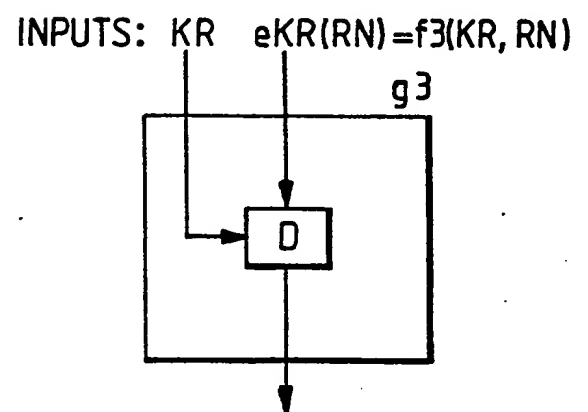


OUTPUT: $eKR(RN) + RN = g3(KR, RN) = K$

FIG. 14 EXAMPLE OF FUNCTIONS $f3$ AND $g3$



OUTPUT: $eKR(RN) = f3(KR, RN)$



OUTPUT: $RN = g3(KR, RN) = K$

FIG. 15 EXAMPLE OF FUNCTIONS $f3$ AND $g3$

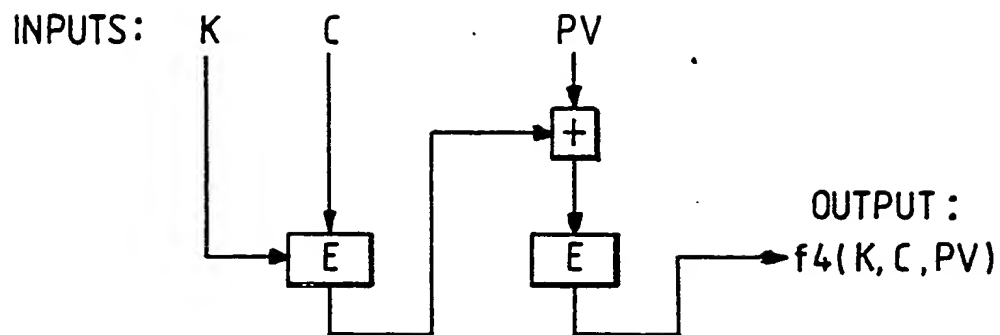
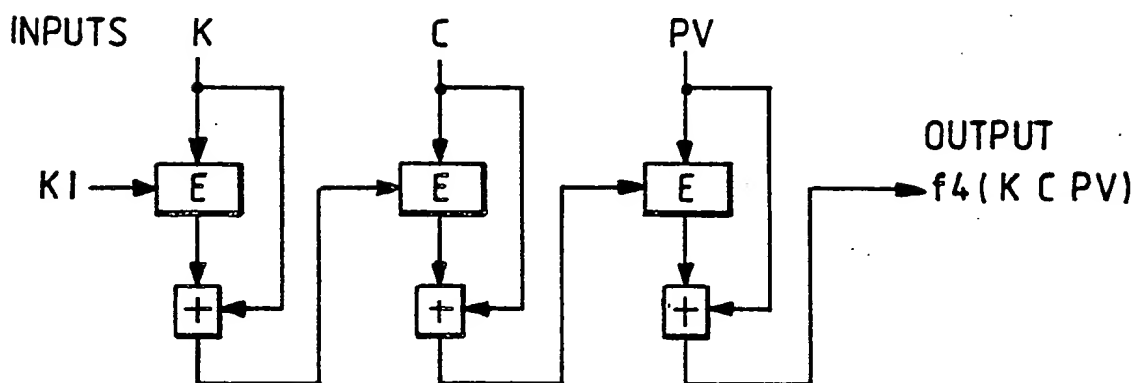


FIG.16 EXAMPLE OF FUNCTION f_4



KI IS A CONSTANT NONSECRET KEY USED BY THE ALGORITHM

FIG. 17 EXAMPLE OF FUNCTION f_4

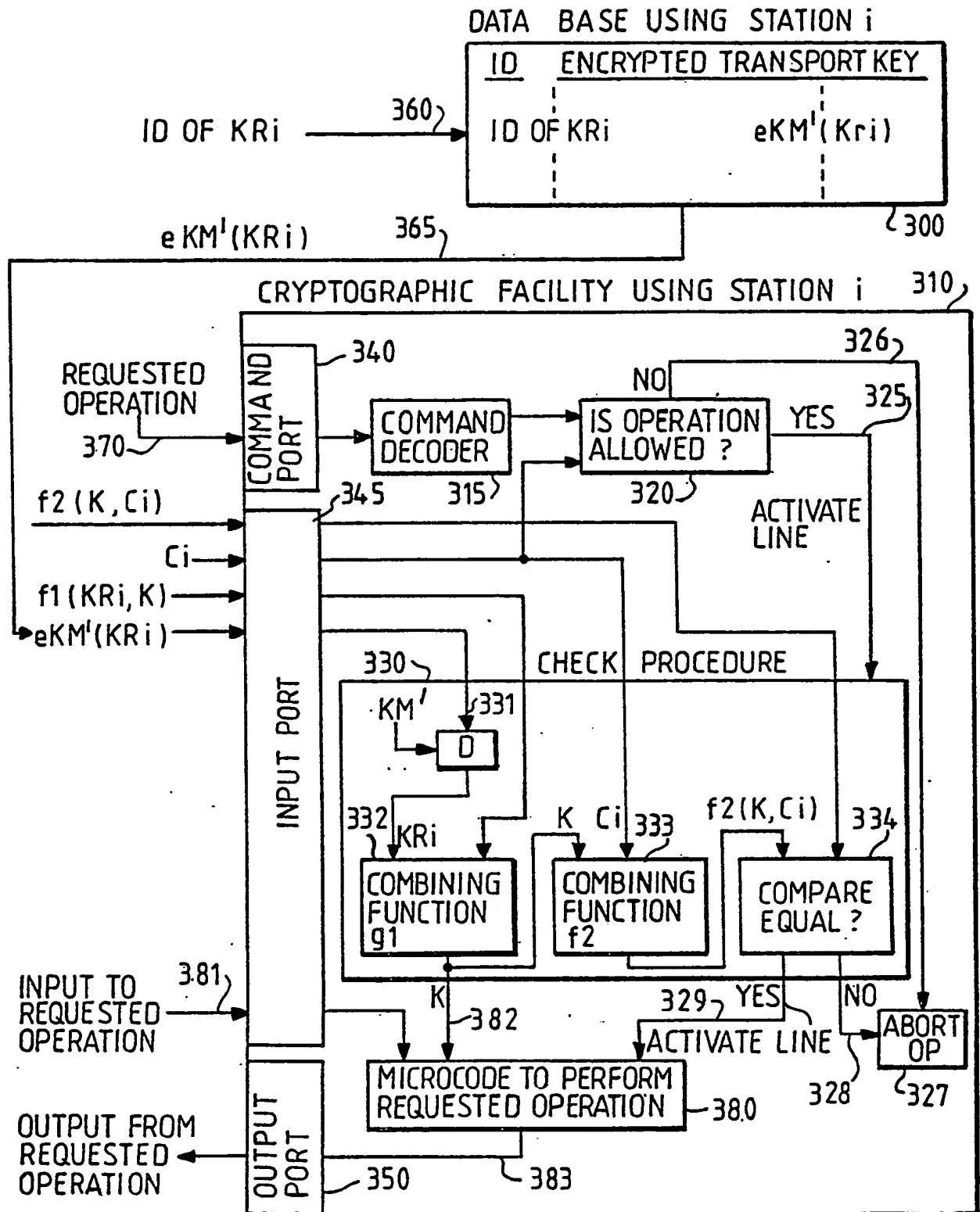


FIG. 18

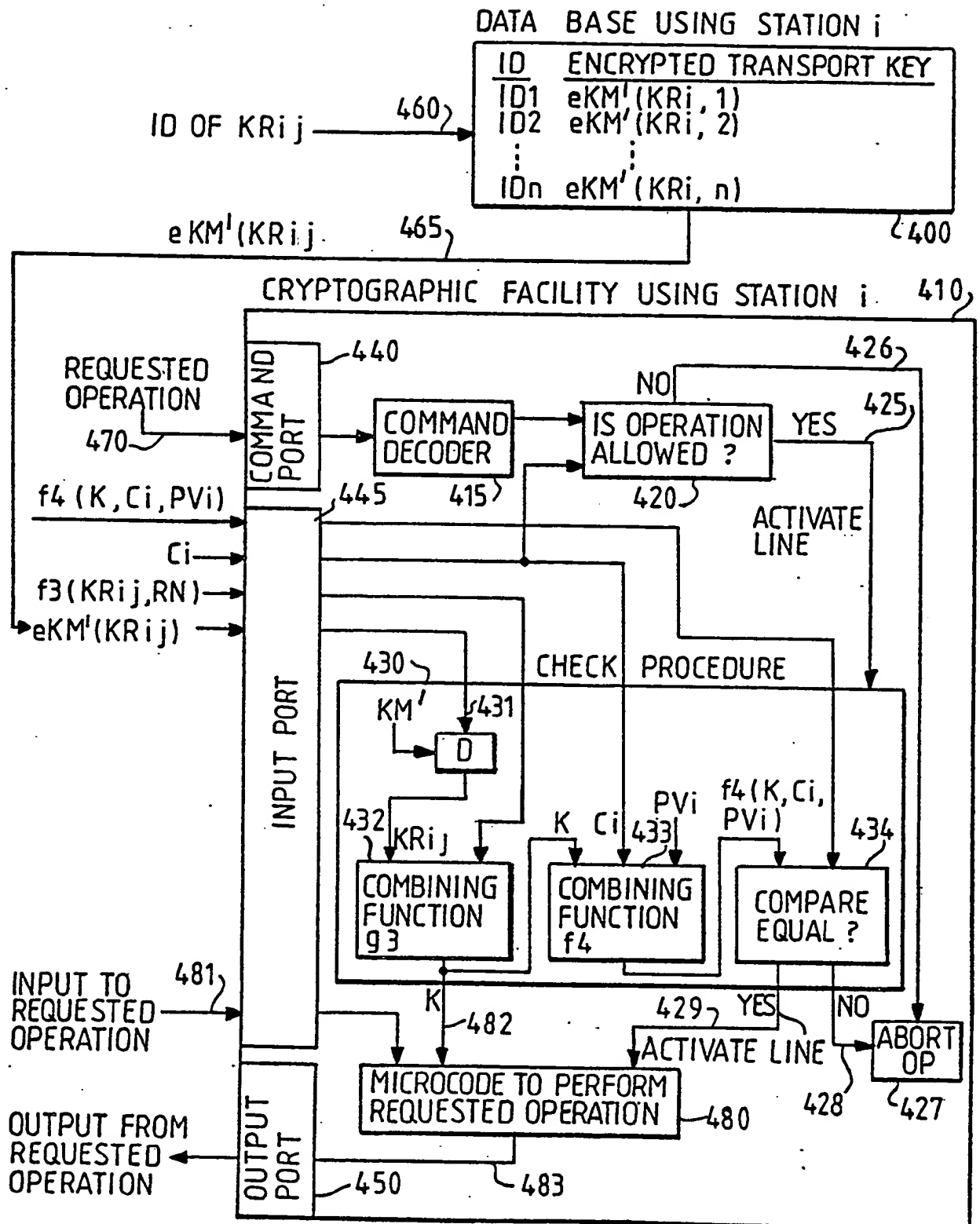


FIG. 19

ID_i OF KEY SHARED BETWEEN
GENERATING STATION AND
USING STATION i; AND
ID_j OF KEY SHARED BETWEEN
GENERATING STATION AND
USING STATION j

DATA BASE AT GENERATING STATION

ID	ENCRYPTED TRANSPORT KEY
ID1	eKM' (KR1)
ID2	eKM' (KR2)
...	...
IDn	eKM' (KRn)

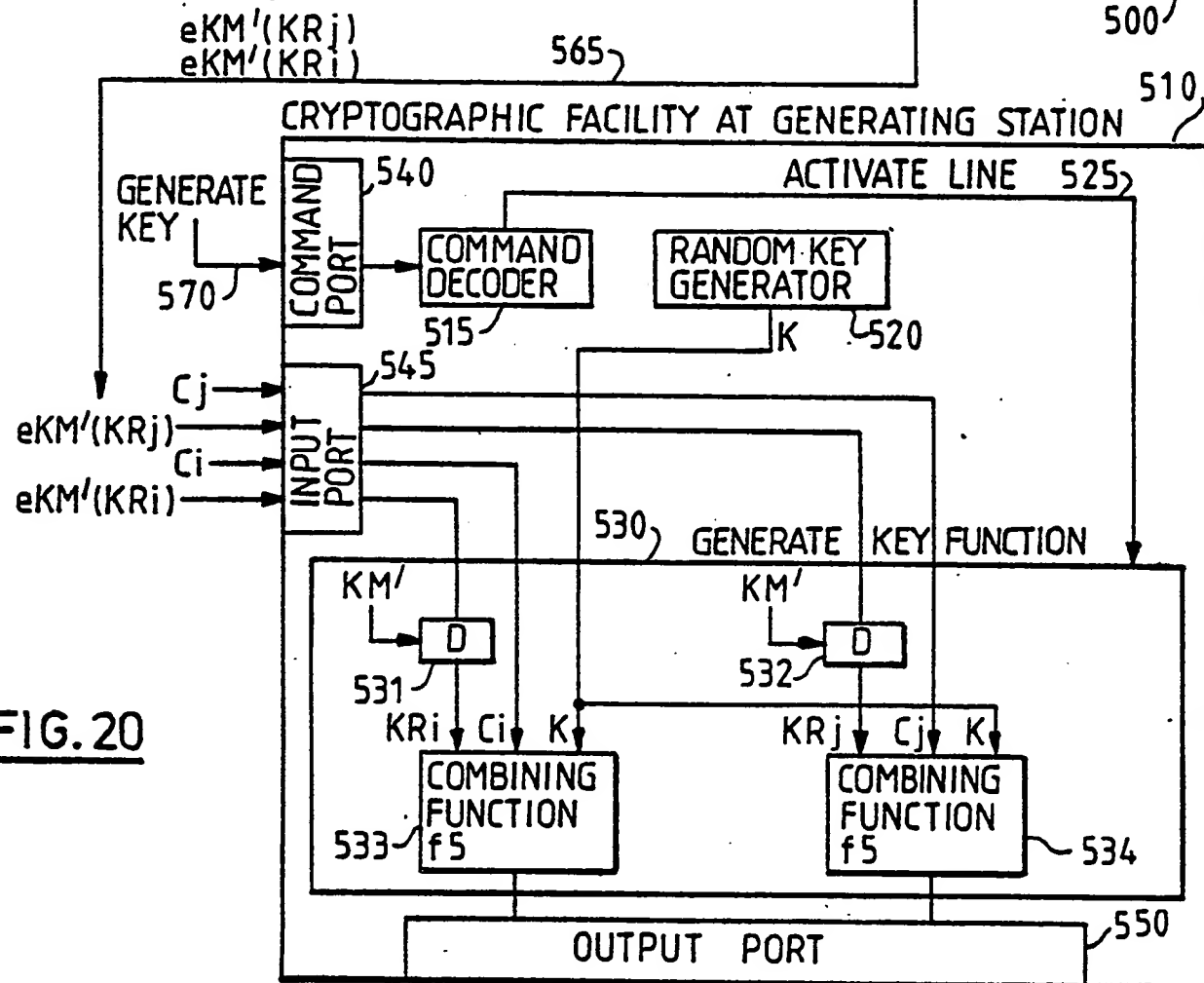
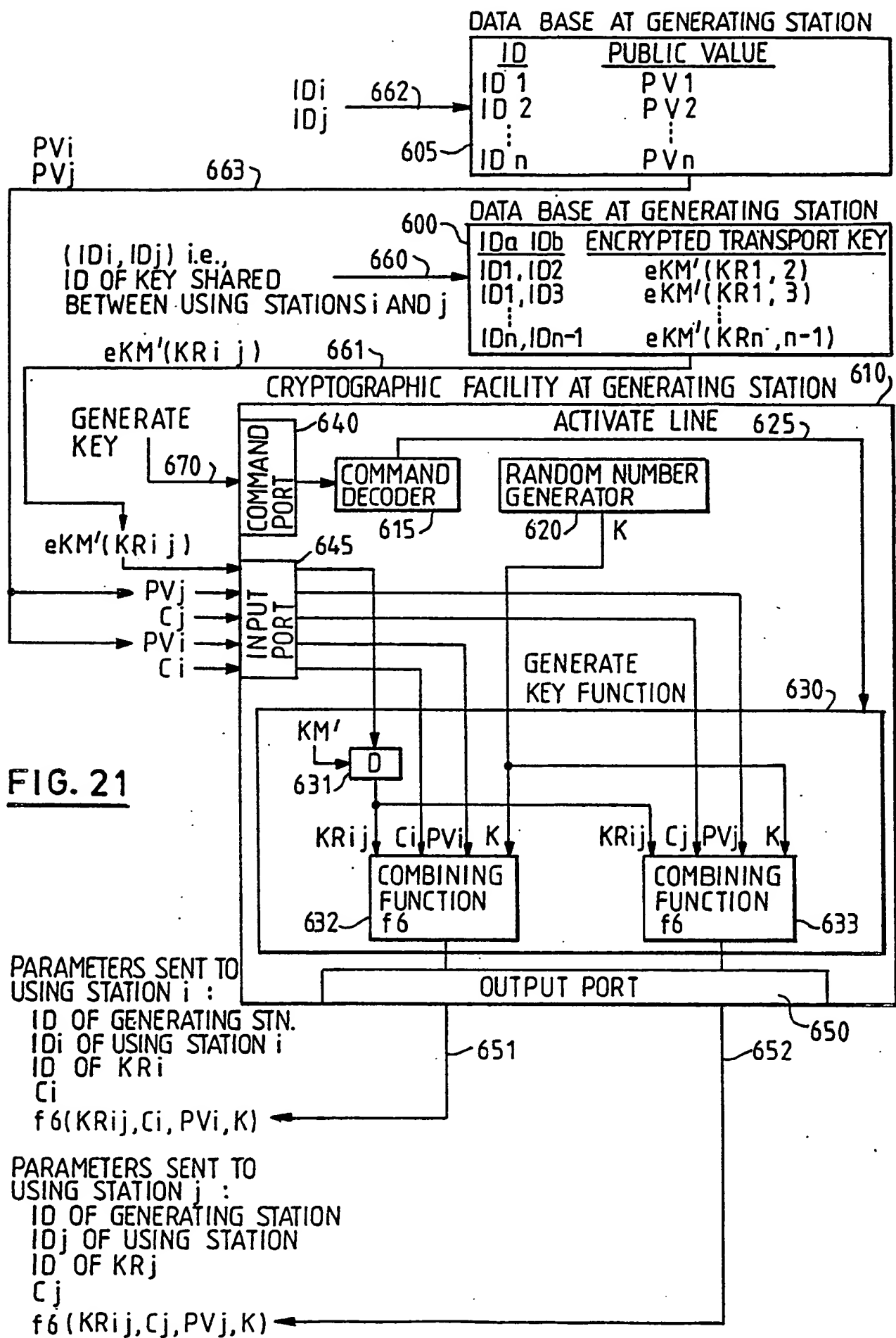
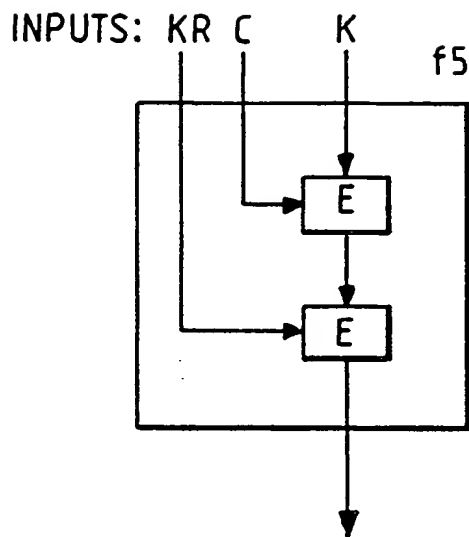


FIG. 20

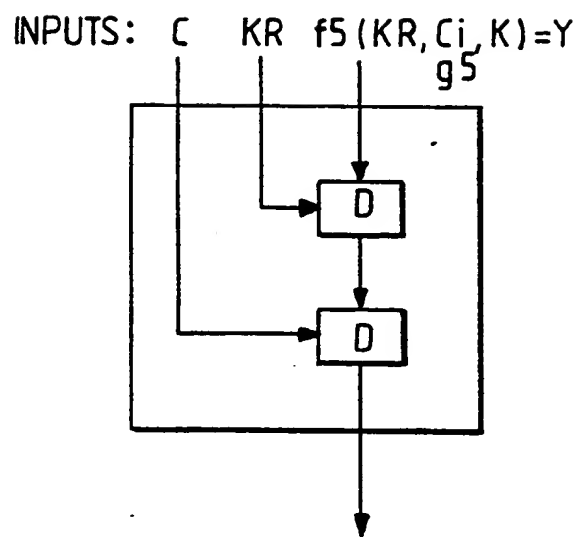
PARAMETERS SENT TO
USING STATION i :
ID OF GENERATING STN.
ID_i OF USING STATION
ID OF KR_i
C_i
f₅(KR_i, C_i, K) ←

PARAMETERS SENT TO
USING STATION j :
ID OF GENERATING STATION
ID_j OF USING STATION
ID OF KR_j
C_j
f₅(KR_j, C_j, K) ←



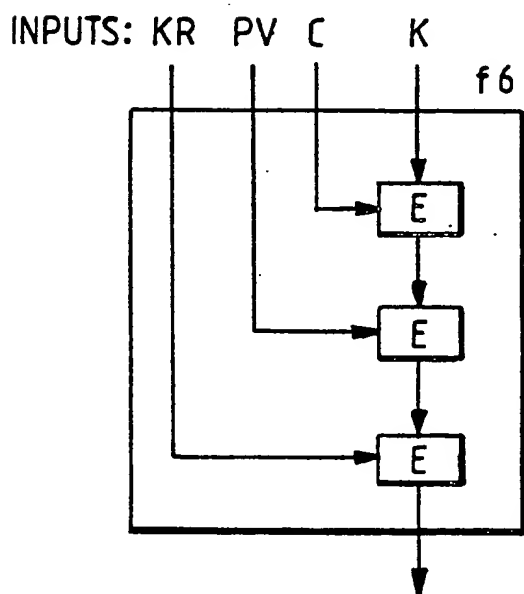


OUTPUT: $f_5(KR, C, K) = Y$

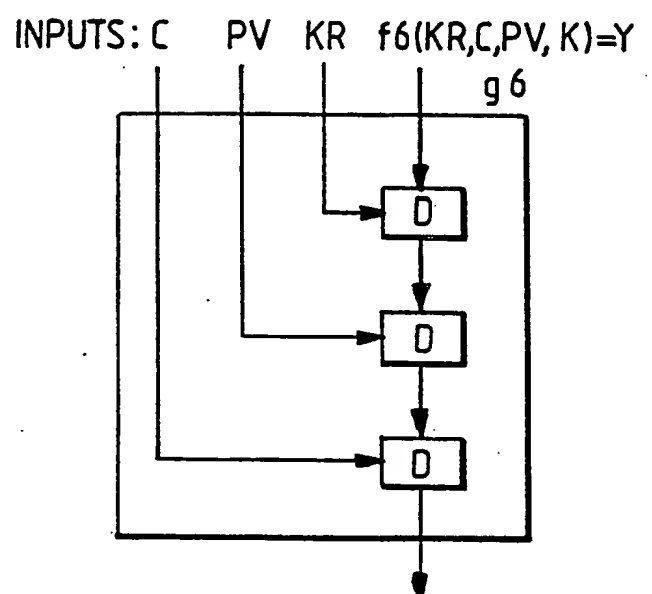


OUTPUT: $g_5(KR, C, Y) = K$

FIG. 22 EXAMPLES OF FUNCTIONS f_5 AND g_5



OUTPUT: $f_6(KR, C, PV, K) = Y$



OUTPUT: $g_6(KR, C, PV, Y) = K$

FIG. 23 EXAMPLES OF FUNCTIONS f_5 AND g_6

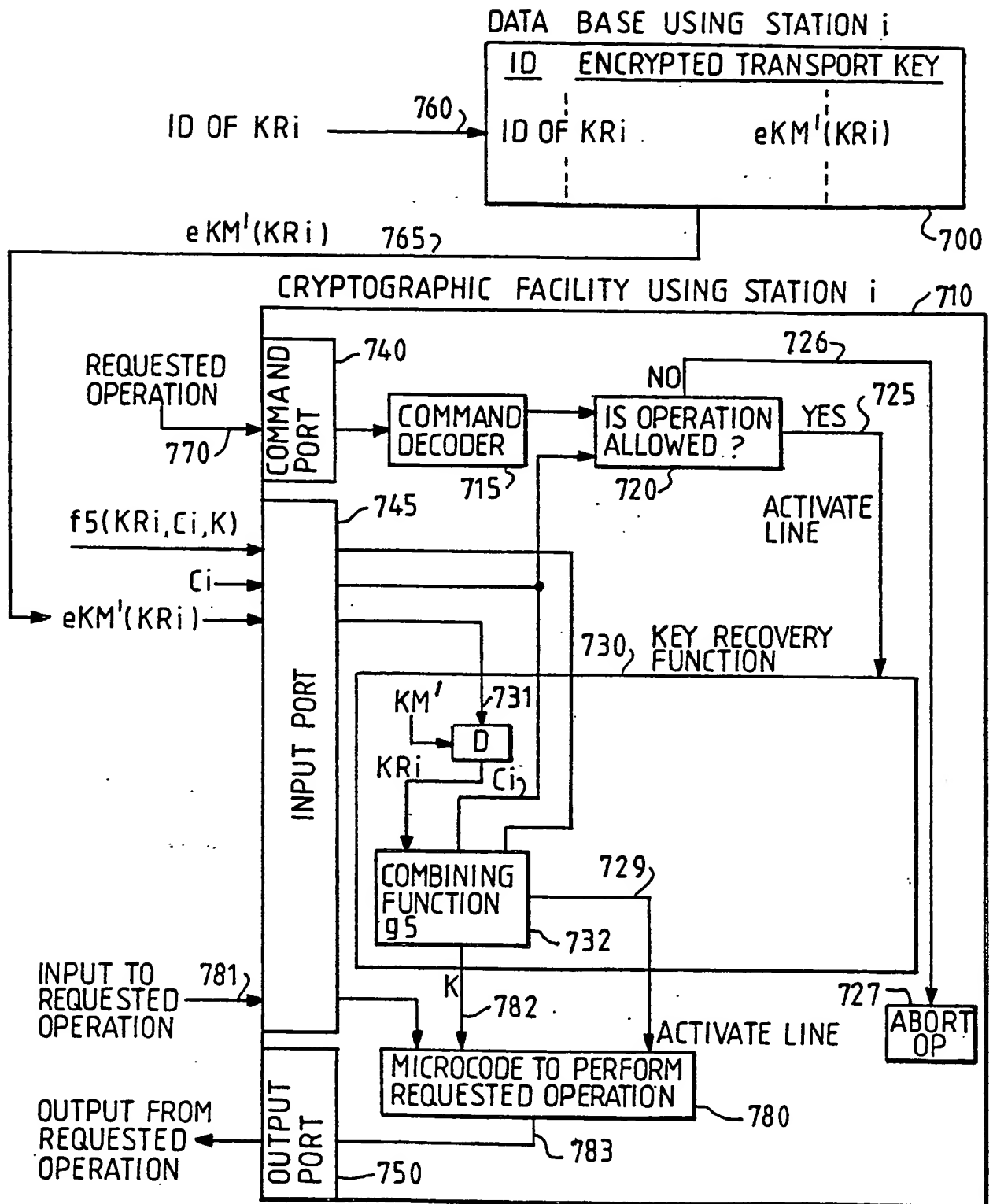


FIG. 24

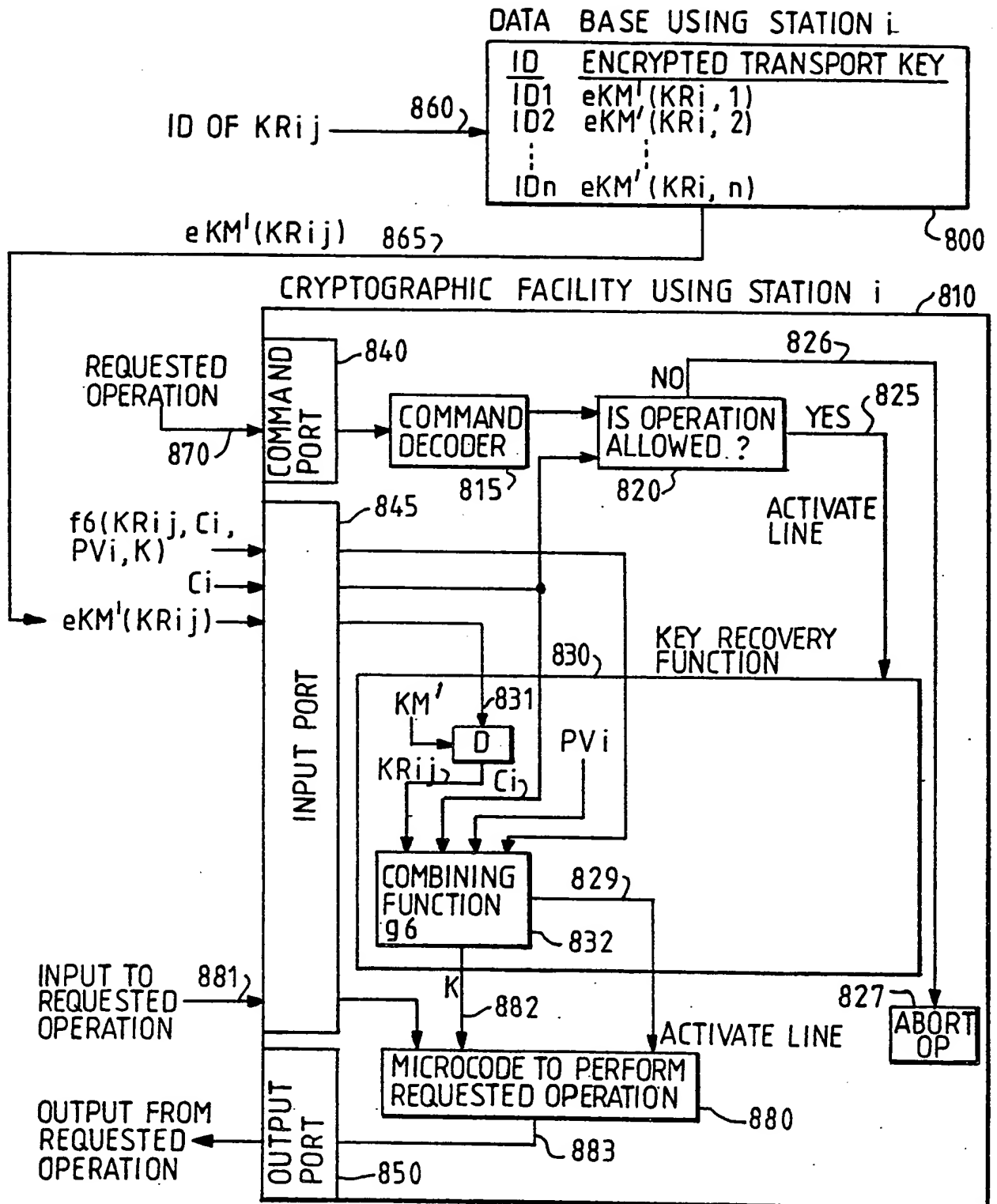
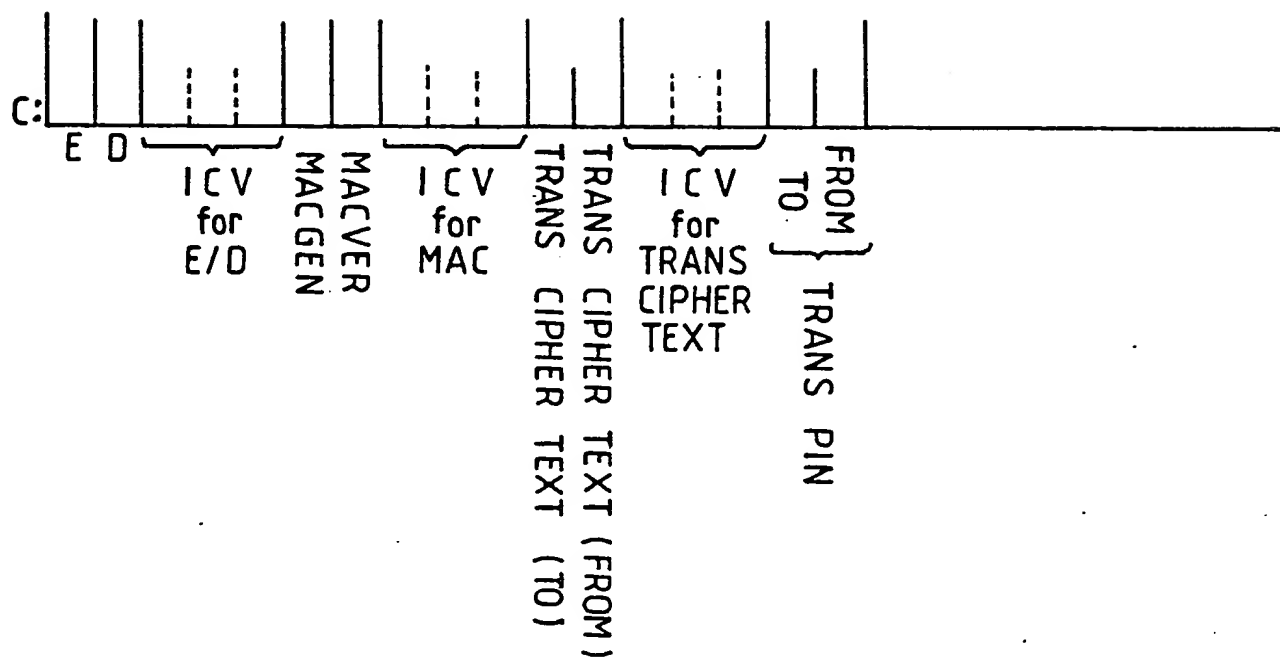


FIG. 25



E : ENCIPHER

D : DECIPHER

ICV : ENCRYPTED ICV.

PLAIN ICV

NO ICV, ie. ICV = 0

FIG. 26

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☒ **BLACK BORDERS**
- ☒ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☒ **FADED TEXT OR DRAWING**
- ☒ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☒ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.